AD-782 706

INTERNAL MULTI-DIMENSIONAL SCALING OF CATEGORICAL VARIABLES

Jeffrey Chit-Fu/Chang, et al

Georgia University
Athens, Georgia

July 1974

AD782706

UNCLASSIFIED

Security Classification

AD-782706

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) UNIVERSITY OF GEORGIA ATHENS, GEORGIA	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP UNCLASSIFIED

3. REPORT TITLE

INTERNAL MULTI-DIMENSIONAL SCALING OF CATEGORICAL VARIABLES

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

JEFFREY CHIT-FU CHANG & ROLF E. BARGMANN

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
JULY 1974	258	42

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S) TECHNICAL REPORT NUMBER 108 DEPARTMENT OF STATISTICS & COMPUTER SCI.
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) THEMIS NUMBER 34
d.	

10. DISTRIBUTION STATEMENT

REPRODUCTION IN WHOLE OR IN PART IS PERMITTED FOR ANY PURPOSE OF THE UNITED STATES GOVERNMENT.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY OFFICE OF NAVEL RESEARCH WASHINGTON, D. C.

13. ABSTRACT

The purpose of the study in this dissertation is to translate raw categorized data into numberical values on which standard statistical analyses can be performed. When raw observations are recorded on a nominal scale, they are to be transformed so that the resulting numbers can be regarded as lying on an interval scale.

A scaling technique is developed on the basis of a generalization of Lancaster's approach (canonical correlation for two sets). In order to generalized the measurement of internal dependence to more than two sets, a single number is needed i.e. $|P|$, the determinant of the correlation matrix. We seek a set of canonical weights $\underline{a}_i$ for all sets $i = 1, 2, \cdots, k$, such that $|P|$ is a minimum; this is Steel's approach (restatement and expansion), and called the Minimum-Determinant approach. The reason for choosing this approach is that there exists a relationship between the minimum likelihood-ratio statistic and maximum likelihood estimates; several examples have been presented. Since $|P|$ is the minimum-determinant (and that is the minimum likelihood-ratio statistic for testing the $H_o : P = I$) the weights $\underline{a}_i$ for all i's will be maximum likelihood estimates of canonical weights.

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*	2a. REPORT SECURITY CLASSIFICATION
	2b. GROUP
3. REPORT TITLE	
4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*	
5. AUTHOR(S) *(First name, middle initial, last name)*	
6. REPORT DATE	7a. TOTAL NO. OF PAGES / 7b. NO. OF REFS
8a. CONTRACT OR GRANT NO. / b. PROJECT NO. / c. / d.	9a. ORIGINATOR'S REPORT NUMBER(S) / 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
10. DISTRIBUTION STATEMENT	
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT cont.

This dissertation also presents computer programs starting from data in contingency tables which are converted into a correlation matrix. Initial values are used in order to start the minimum-determinant process. Various initial weights and the final minumum-determinant solution have been compared. A good initial guess for the canonical weights is the "canonical-multiple correlation" approach (the characteristic vector associated with the largest characteristic root -- the square of the canonical-multiple correlation of one set vs. the totality of the others). If high accuracy of the estimate of canonical weights is desired, the Fletcher and Powell process can be used to obtain the minimum-determinant solution.

Four numerical examples have been presented and a validation study demonstrates the qualifty of results.

In the appendices, listing of computer programs and input card layout have been given.

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
CANONICAL WEIGHTS						
MINIMUM-DETERMINANT SOLUTION						
CANONICAL-MULTIPLE CORRELATION						
MINIMUM LIKELIHOOD-RATIO STATISTIC						
SCALING						
MULTI-DIMENSIONAL SCALING						
DEPENDENCE IN MULTIPLE SETS						
MULTIVARIATE ANALYSIS						

THEMIS TECHNICAL REPORT NUMBER 34

TECHNICAL REPORT NUMBER 108

INTERNAL MULTI-DIMENSIONAL SCALING OF CATEGORICAL VARIABLES

by

JEFFREY CHIT-FU CHANG & ROLF E. BARGMANN

Reproduction in whole or in part is permitted for any purpose of the United States Government. This Research was supported, in part, by the Office of Naval Research, Contract Number

N00014-69-A-0423, NR 042-261

Rolf Bargmann
Principal Investigator

The University of Georgia

Department of Statistics and Computer Science

Athens, Georgia

July 1974

TABLE OF CONTENTS

Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

1.1 Introduction

In an article in Science, S. S. Stevens [1946] suggested some difinitions of scales for observations, which have become rather widely adopted. Most statistical analyses, especially multivariate methods, require that measurement be available in an "interval scale" (in Stevens' terminology), i.e. that distance from a point on the scale to another can be related to distance between two points at a different location of the scale. The weaker assumption that data be on an "ordinal scale" presents no serious problem either. Rank ordering methods can be applied, or empirical transformations can be made from such ordinal scales to marginally normal interval scales; methods to do this were widely used in the late 19th century, especially in Psychophysics.

The last of Stevens' scales, the "nominal scale", poses quite different problem. A variable possessing a nominal scale can be formally translated into numbers, but such numbers serve for identification only. For example, the variable "color of hair" could be recorded as blond = 1; brown = 2; black = 3; red = 4; etc. It would obviously be absurd to obtain "mean values", linear combination, or

"standard deviations" of such numbers. But what is one to do if a study of relationship between, say, eye color and hair color (and other nominally scaled variables) is desired?

The object of scaling is the translation of raw data into some other numerical values so that standard statistical analyses may be performed on the latter. The standard statistical analyses - paired and pooled t tests, analysis of variance, etc. - assume that the data are scored on an interval scale. Where the raw data are in the form of ranks, or in the form of grouped ordinal data, analyses are often adequate even if no transformations are made. For example, in the Kruskal-Wallis technique, ranks are treated as if they were numbers on an interval scale. Only where the number of sequential categories is very small, or where the proportion of data in each category is far removed from the expected proportion under a normal distribution, is it necessary to apply one of the marginal normalization techniques.

Where raw observations are recorded on a nominal scale, the object of scaling is to transform them so that the resulting numbers can be regarded as lying on an interval scale. Such translation is impossible unless some additional information is available - criterion groups or, as in the present case, information on other nominal variables. As some form of distribution must be approximated in such scaling, the normal distribution is usually chosen as the one to be approached by the transformed data. The reason for choosing

the normal distribution, univariate or multivariate, is that the various forms of linear analyses lead to "best" test statistics when errors are additive and normally distributed. According to Gauss, when errors are additive and when linear estimators are the maximum likelihood estimators of the location parameters ("Axiom of the Mean") the errors will have a normal distribution, provided only that data be continuous, and that there be three or more observations.

For scaling of nominal variables it became clear to earlier workers that some criterion, some principle, must be utilized to transform the category numbers or levels in a nominal scale into new scale values which have, approximately the properties of interval scales. In contrast to Stevens' terminology, such variables have been called "categorical" or "categorized", in the literature (M.G. Kendall [1948]); we shall employ that designation. R.A. Fisher [1940] proposed a set of weights (i.e. numbers into which the original nominal values are to be translated) chosen in such a way that, in terms of the new scaled values, Euclidean distances between some well defined criterion groups become as large as possible. Lancaster [1957] regards the criterion variable also as a random variable, and shows that a scaling procedure based on this criterion variable gives the closest approach to bivariate normality. However, the same author [1960] then proceeds in a direction of stepwise regression rather than multivariate normality, and thus has merely an approximation to

the more general problem of several sets of categorized variables. The bivariate methods proposed by Fisher and Lancaster produce identical scales under all conditions; they are, in modern terminology, discriminant functions for dummy variables $y_1, y_2, \ldots, y_k$, where $y_i = 1$ if the categorical scale value i has been applied to an observation, and 0 otherwise. The equivalence of this technique with Lancaster's canonical correlation approach was used by Kundert and Bargmann [1972] in order to scale each categorical variable against several criteria.

A multivariate generalization of Lancaster's approach is not available. Such an approach is needed when it is impossible to classify categorized variables into criteria and responses, where, in fact $(k - 1)$ variables would have to serve as criteria for the k^{th} in a set of k nominal variables. In this dissertation, a scaling technique is developed on the basis of such an approach. Frequencies of occurrence of each combination of k values are recorded in a k-dimensional contingency table (see CHAPTER IV). Experimental units occurring in the same cell of the k-way contingency table have the same vector of categorical responses $[c_1, c_2, \ldots, c_k]$, where the c_j are integers. In analogy with Lancaster's model, k sets of dummy variables are constructed, each set having as many members as the categorical variable has levels. From this information, a super-matrix can be constructed, consisting of $k(k + 1)/2$ submatrices R_{ij}, where R_{ij} contains

the sample correlations between the dummy variables in the i-set and those in the j-set. The internally-scaled variables are then found, in much the same way as the approaches by Fisher and Lancaster, as canonical weights on each of the k sets of dummy variables. To solve this problem, however, we must make a choice, based on the problem on hand, among the many proposed generalization of canonical correlation ; we chose the Minimum-Determinant criterion proposed by R.G.D. Steel [1949] , i.e. the set of weights which minimizes the determinant of the resulting correlation matrix. The reasons for the preference are stated, in some detail, in section 3.2. To be usable in conjunction with likelihood-ratio testing, Steel's results must be restricted to the correlation matrix. The determinant of the variance-covariance matrix, inappropriately called "generalized variance", has no meaning in likelihood-ratio testing.

1.2 Literature Review

H.Hotelling [1936] proposed the canonical correlation as a measure of dependence between two sets of random variables. The definition has no distributional assumptions; however, the use of linear combinations restricts the distribution to classes which are closed under linear operations; there are very few such classes (multivariate normal and Dirichlet, for example). From a set of response variables $\underline{y}$, a single variable $u = \underline{\alpha}'\underline{y}$ is formed and from a set $\underline{z}$, a

linear combination $v = \underline{\beta}'\underline{z}$ can be formed where $\underline{\alpha}$ and $\underline{\beta}$ are chosen in such a way that $|\mathrm{corr}(u, v)|$ is a maximum. An elementary derivation shows that this "canonical correlation" is the positive square root of the largest characteristic root of a matrix product $\Sigma_{yy}^{-1}\Sigma_{yz}\Sigma_{zz}^{-1}\Sigma_{yz}'$, where Σ_{yy} denotes the variance-covariance matrix for the $\underline{y}$ set, Σ_{zz} that for the $\underline{z}$ set and Σ_{yz} the covariance matrix whose (i,j) element is the covariance between y_i and z_j. The ideal weights $\underline{\alpha}$ and $\underline{\beta}$ are called "regression-like" parameters (S.N. Roy [1957]) and "weights of the best-predictable criterion" (Hotelling [1936]), respectively. Given observations obtained from a sample, sample covariance or correlation matrices can be employed to obtain the maximum likelihood estimates r^2, $\underline{a}$ and $\underline{b}$ of the above-mentioned parameters ρ^2, $\underline{\alpha}$ and $\underline{\beta}$, for the multivariate normal case. The sample canonical correlation coefficient is a test statistic (based on the "Union-Intersection" principle) for the test of independence between two sets of random variables.

Although S.S. Wilks [1935] obtained likelihood-ratio test statistics for the test of independence in two (and several) sets of random variables, the sample canonical correlation, which was discovered later, does not yield the same test; where the latter is the largest characteristic root of $R_{yy}^{-1} R_{yz} R_{zz}^{-1} R_{yz}'$, the likelihood-ratio statistic is $|I - R_{yy}^{-1} R_{yz} R_{zz}^{-1} R_{yz}'|$, i.e. the product of the complements

of all the roots. The same author [1932] also found the likelihood-ratio statistic for the test for one set $\underline{y}$, of internal independence, to be $|R|$, the sample correlation matrix of the $\underline{y}$ set. He also found that the statistic to test $H_0: \sum = I$ is $|S| e^{-tr\ S}$ but, unfortunately, called the meaningless first factor of this expression the "generalized variance".

In his dissertation, R.G.D. Steel [1949] studied the problem of generalizing the canonical correlation to k (>2) sets of variables but this problem should not be confused with the likelihood-ratio test statistic for independence in k-sets, which was described by Wilks [1935] and Wald-Brookner [1941] . Steel's approach consisted in constructing k sets of weights $\underline{\alpha}_1', \underline{\alpha}_2', \ldots , \underline{\alpha}_k'$ to be applied to the k sets of random variables $\underline{y}_1, \underline{y}_2, \ldots , \underline{y}_k$ in such a way that if $u_i = \underline{\alpha}_i' \underline{y}_i$ $(i = 1, 2, \ldots , k)$, the determinant of the matrix of correlations between the u_i's is a minimum. Steel thus appears to be the first author to estimate parameters in such a way that the resulting likelihood-ratio statistic (here the correlation determinant, the L. R. statistic for the test of internal independence) is maximized or minimized. This principle has been applied to advantage by several later authors.

A single Union-Intersection test statistic for the test of independence has not been described. Roy and Bargmann [1958] proposed a set of several statistic ("step-down

correlation") for this purpose. For estimation purposes, however, a single criterion function is required. Social scientists have described a multitude of indices to express the relationship between k sets of variables by a single number; however, since such indices are invariably defined for a sample only (and not, as they must be, as parametric functions) they are of no value for statistical inference and must be regarded as descriptive statistics only.

P. Horst [1961] offers four different suggestions for indices of correlations among several sets of variables. As in the previous papers by him and others, these indices are defined for samples only. Initially, Horst performs the same reduction as Steel [1949], (an algorithm which is explained in section 4.1) in order to obtain identity matrices for the correlations within each set. In one of the cases, a weighted sum of the correlations is employed. In another index, the matrices are approximated by matrices of rank one. "Maximum variance" (actually, weighted sums of squares) methods are suggested on correlation elements. To some social scientists who report numerous indices in their studies such an index number may have some relative meaning (just as the Dow-Jones index does to ecoromists, or the Intelligence-Quotient to some educators).

Such numbers cannot be used as criteris for statistical estimation or inference, since they have no relation to estimable parameters. As demonstrated in section 4.1, Steel's

Minimum-Correlation-Determinant seems to be the only currently known generalization that is based on a parametric representation, and can therefore be used for the estimation of weights.

The following treatments of the nominal scaling problem constitute a more comprehensive approach. Again, a choice had to be made from an abundance of material (e.g. the different, often contradictory, approaches scattered throughout Vol. II: Inference and Relation, of Kendall and Stuart [1961]).

R.A. Fisher [1949] regarded each categorical variable as k (zero-one) variables ($y_i = 1$ if the response was i, 0 otherwise). His scale values are what we would today call the estimated weights of the discriminant function, i.e. those that produce maximum discrimination between defined groups according to a single criterion. He proposed an iterative solution rather than the simpler characteristic root and vector approach which is used today.

H.C. Lancaster [1957] obtained the same results as R.A. Fisher, using characteristic root-characteristic vector methods and starting from different assumptions. However, his objective was to approximate a pair of categorical variables by a bivariate normal pair of variables. For Lancaster [1957] the "criterion" of R.A. Fisher is itself a categorical variable. Thus, Lancaster has two sets of dummy variables: $\underline{y}' = [y_1, y_2, \ldots, y_r]$ for the r levels of categorical variable y, and $\underline{z}' = [z_1, z_2, \ldots, z_s]$

for the s levels of categorical variable z. An experimental unit which has response i on categorical variable y and response j on categorical variable z would thus have s + r responses on the dummy variables with $y_i = 1$, $z_j = 1$ and all other dummy variables equal to zero (of course the resulting correlation matrices are singular). Lancaster then shows that, if scale values $\underline{a}$ are chosen for $\underline{y}$ and scale values $\underline{b}$ are chosen for $\underline{z}$ so that $|\text{est.corr}(\underline{a}'\underline{y}, \underline{b}'\underline{z})|$ is a maximum, then the scaled variates have a best approximation to normality. In a later attempt to generalize the method to k categorical variables, Lancaster [1960] uses different arguments, which are not as convincing as the method used for the scaling of two nominal variables.

Hays [1963] suggests a quantity η^2, to measure the ability of a predictor to explain the variance of each dependent variable code dichotomized against the others. A natural generalization of this across codes would be the ratio of the sum of with-group sums of squares to the sum of total sums of squares, i.e.

$$\eta_i^2 = \frac{\Sigma V_{ip}}{\Sigma T_p} \quad \text{or} \quad \eta_i^2 = \frac{\Sigma S_p^2 \eta_{ip}^2}{\Sigma S_p^2}$$

where

$V_{ip} = \sum_j w_{ij}(y_{ij} - \bar{y}_p)^2$, within-group sums of squares for the p^{th} dummy dependent variable on i^{th} predictor.

$T_p = \sum_k w_k(y_{kp} - \bar{y}_p)^2$, total sums of squares for the p^{th} dummy dependent variable.

$$S_p^2 = \frac{\sum_k w_k (y_{kp} - \bar{y}_p)^2}{\sum_k w_k}$$ sample variance of p^{th} dummy variable.

$w_{ij} = \sum_{k \in Q_{ij}} w_k$ weight sum for j^{th} code of i^{th} predictor.

w_k sampling weight for $k \in Q_{ij}$.

Q_{ij} subset of individual having j^{th} code value on predictor i.

$\eta_{ip}^2 = \frac{V_{ip}}{T_p}$ ratio of within-group sum of squares to the total sum of squares on p^{th} dummy variable on i^{th} predictor.

Messenger and Mandell [1972] suggest a bivariate θ_i to measure strength of association using a criterion of correct placement in the dependent variable code which is a linear transformation of the Goodman-Kruskal Lamda statistic (Goodman and Kruskal [1954]). They claim that it has more intuitive appeal than Lamda and fits more naturally with a multivariate model. Theta is defined simply as the proportion of the sample correctly classified when using a prediction-to-the-mode-strategy in the frequency distribution of each category.

For multivariate cases, <u>two</u> statistics are used to

measure the multivariate strength of association. These are the generalized squared multiple regression coefficient R^2 and a multivariate version of ∂ , the Theta statistic. This statistic generalizes the bivariate prediction-to-the-mode concept. It is defined as the proportion correctly classified using a decision rule that assigns each individual to that dependent variable category which has the maximum forecast value for that individual; this latter principle is, of course, similar to R.A. Fisher's. It appears that the Messenger and Mandell technique bears the same relation to the Fisher-Lancaster technique as the step-wise 0-1 multiple regression approximation bears to the discriminant function.

In an attempt to define single indices of correlation among k sets, J.McKeon [1962] starts with the correlation between two measurements x and y, based on a paired sample of size n. He then defines a generalized measure of product moment correlation among k sets of variates by

$$\rho(x_1, x_2, \ldots, x_k) = \max r_I(a_1x_1, a_2x_2, \ldots, a_kx_k)$$

$$= \frac{2}{n-1}\left[\frac{\sum_{i<j} a_i a_j \ SP(x_i, x_j)}{\Sigma a_i^2 \ SSx_i}\right] \qquad (1.2.1)$$

where

r_I is the intraclass correlation

a_i is an arbitrary set of weights, each $SP(x_i, x_j)$ is a sum of products, and each SSx_i is a sum of

squares.

He then defines the "Generalized Canonical Correlation" for k sets of variates as the maximum value of the generalized product moment correlation for the k linear composites $\underline{a}_i' \underline{x}_i$ $(i = 1, 2, \ldots, k)$, with respect to variation of the a_i. Then

$$\rho(x_1, x_2, \ldots, x_k) = \max r_I(a_1'x_1, a_2'x_2, \ldots, a_k'x_k)$$

$$= \max \frac{1}{k-1}\left[\frac{\Sigma a_i' S_{ij} a_j}{\Sigma a_i' S_{ii} a_i} - 1\right] \qquad (1.2.2)$$

where

$$S_{ij} = \sum_{\alpha}(\underline{x}_\alpha - \overline{\underline{x}})(\underline{y}_\alpha - \overline{\underline{y}})' \qquad (1.2.3)$$

$$\underline{x}_\alpha' = [x_{1\alpha}, x_{2\alpha}, \ldots x_{p\alpha}]$$

$$\underline{y}_\alpha' = [y_{1\alpha}, y_{2\alpha}, \ldots y_{q\alpha}]$$

$(i = 1, 2, \ldots, p$ variates)

$(j = 1, 2, \ldots, q$ variates)

$(\alpha = 1, 2, \ldots, n$ observations)

This is equivalent to maximizing the quantity

$$\gamma = \frac{\Sigma a_i' S_{ij} a_j}{\Sigma a_i' S_{ii} a_i} = (k-1)r_I + 1 \qquad (1.2.4)$$

Let S be the sum of products matrix for the k sets combined and let S_d be the diagonal super-matrix with element S_{ii}.

Let $\underline{a}' = [\underline{a}_1', \underline{a}_2', \ldots \underline{a}_k']$ be the vector of combined

weights, then

$$\gamma = \frac{\underline{a}' \, S \, \underline{a}}{\underline{a}' \, S_d \underline{a}} \tag{1.2.5}$$

As usual, the maximizing γ and $\underline{a}$ satifies the relation

$$(S_d^{-1} \, S - \gamma \, I) \, \underline{a} = \underline{0} \quad , \quad r_I = \frac{\gamma - 1}{k - 1} \tag{1.2.6}$$

and $\quad \gamma = Ch_{max}(S_d^{-1} \, S)$ (1.2.7)

where Ch_{max} denotes the largest characteristic root.

He extends canonical correlation to more than two sets of variables, based upon a generalized association measure.

$$r_I = \frac{2}{k-1} \left[\frac{\sum\limits_{i<j} \sigma_{ij}}{\sum\limits_{i} \sigma_i^2} \right] = \frac{\sigma_t^2 - \sum\limits_i \sigma_i^2}{(k-1) \sum\limits_i \sigma_i^2} \tag{1.2.8}$$

where σ_{ij} are the covariance of variables.

σ_t^2 are the variance of their sums. [1)]

He also discussed another possible generalization from a maximization of

$$h = \frac{\sum\limits_{i<j} \sigma_{ij}}{\Sigma \sigma_i \, \sigma_j} = \frac{\sigma_t^2 - \Sigma \, \sigma_i^2}{(\Sigma \, \sigma_i)^2 - \Sigma \, \sigma_i^2} \tag{1.2.9}$$

h has a solution in terms of roots and vectors for the case of a single variate per set, and this is closely related to Lovinger's [1947] coefficient of homogeneity.

---

1) This is McKeon's notation, although what he calls σ_{ij} are actually sample quantities.

In dealing with the problem of weights in the absence of a criterion, some authors describe sample index numbers which assign weights to each standardized variable according to its loading on the first principal component of the resulting correlation matrix (Horst [1936] , Edgerton and Kolbe [1936] , Wilks [1938] , Lord [1958]).

Kundert and Bargmann [1972] use the similarities between the Fisher and Lancaster techniques to derive a series of test statistics and their distributions in the case of bivariate categorical scaling. As was stated earlier, these techniques produce identical scale values; however, Kundert and Bargmann use other interpretive statistics (correlations against discriminant function) to identify those levels of a categorical variable which contribute most strongly to its association with some criterion. The main objective of their scale analysis is to try to interpret the dependence between criteria and categorized variables.

In the same general spirit, F.M. Andrew and R.C. Messenger [1973] , in their multivariate Nominal Scale Analysis, stated that a general goal of multivariate data analysis is to understand how a dependent variable is affected by a set of independent variables. They raise five general questions: (1) as a whole, how well do the independent variables explain the variability in the dependent variables? (2) what is the relationship of a particular independent variable to the dependent variables, while other independent

variables are held constant? (3) to what extent the dependent variables can be explained by a particular independent variable, over and above the other independent variables? (4) taking into account a subject's scores on an independent variable, what score should one predict on the dependent variable, and (5) what is the deviation of the prediction from an obserable score?

The Multivariate Nominal Scale Analysis is designed to handle problems where (a) the dependent variable is a set of mutually exclusive categories, (b) the independent variables may be observed at any level, and (c) any form or pattern of relationship may exist between any independent variables and dependent variable and also between any pair of independent variables.

The method is designed to be relevant for "theoretical-oriented" and "conceptual-oriented" analysis. A second characteristic of Multivariate Nominal Scale Analysis is its ability to analyse relative large number of predictors with moderate sized data sets. A third characteristic is its focus on the magnitudes of relationships rather than the statistical significance of those relstionships.

Finally, both the one-way analysis of variance eta-square statistic η^2_{ip} [2] by Hays [1963] and bivariate Theta by Messenger and Mandell [1972] are used to measure

---

2) This is a traditional name given to SSH/(SSH+SSE), the incomplete Beta statistic derived from F.

the strength of the bivariate relationship between the dependent variables and each predictor.

P.H. Dubois [1957] and E. Jenning [1965] defined semi-partial correlation and multiple semi-partial correlation (extended to the canonical semi-partial correlation) when the third set is partialed out from only one of the two sets.

C.A. Smith [1953] has shown that the generalization of the "intraclass" correlation to p measurement on n groups is the largest characteristic root and the associated vector of a certain matrix.

CHAPTER II

CANONICAL WEIGHTS IN CATEGORICAL SCALING

2.1 Relation Between Two Sets

In this section, the well-known results of canonical correlation analysis (analysis of dependence between two sets) will be summarized.

Let

$$\underline{y}' = \left[y_1, y_2, \ldots, y_p\right] \quad , \quad \underline{z}' = \left[z_1, z_2, \ldots, z_q\right] \qquad (2.1.1)$$

and then form new variables as linear combinations of all y's and all z's,

$$u = \underline{\alpha}'\underline{y} \quad , \quad v = \underline{\beta}'\underline{z} \qquad (2.1.2)$$

Choose the vectors $\underline{\alpha}$ and $\underline{\beta}$ in such a way that corr(u.v) is a maximum, and call this maximum the canonical correlation between the sets $\underline{y}$ and $\underline{z}$. If $\underline{y}$ is considered to be the predictor set then call the elements of $\underline{\alpha}$ "regression-like" parameters and if $\underline{z}$ is thought of as the criterion set, then call $\underline{\beta}'\underline{z}$ the "best-predictable-criterion" (Hotelling [1936])

Now $\operatorname{cov}(u,v) = \operatorname{cov}(\underline{\alpha}'\underline{y}, \underline{z}'\underline{\beta}) = \underline{\alpha}' \operatorname{cov}(\underline{y}, \underline{z}')\ \underline{\beta}$

$$= \underline{\alpha}' \textstyle\sum_{12} \underline{\beta} \qquad (2.1.3)$$

where $\sum_{12}$ is the upper right block of the covariance matrix and $\underline{\alpha}$, $\underline{\beta}$ are the weights of the canonical variables $\underline{y}$ and $\underline{z}$.

$$
= \begin{array}{cc} & \begin{array}{cc} (\underline{y}') & (\underline{z}') \end{array} \quad 1) \\ \begin{array}{c} (\underline{y}) \\ (\underline{z}) \end{array} & \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12}' & \Sigma_{22} \end{bmatrix} \begin{array}{c} (p) \\ (q) \end{array} \\ & \begin{array}{cc} (p) & (q) \end{array} \end{array} \quad , \qquad (2.1.4)
$$

Since the length of $\underline{\alpha}$ and $\underline{\beta}$ are indeterminate, they can be chosen so that

$$\underline{\alpha}'\Sigma_{11}\,\underline{\alpha} = 1 \quad , \quad \underline{\beta}'\Sigma_{22}\,\underline{\beta} = 1 \qquad (2.1.5)$$

hence var(u) = 1 and var(v) = 1, therefore, under the above constraints, corr(u,v) = cov(u,v) = $\underline{\alpha}'\,\Sigma_{12}\,\underline{\beta}$.

It is well-known that the square of this maximum correlation, is the largest characteristic root of

$$\Sigma_{11}^{-1}\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}',$$

and $\underline{\alpha}$ is the associated characteristic vector; $\underline{\beta}$ is easily found to be

$$\underline{\beta} = \Sigma_{22}^{-1}\,\Sigma_{12}'\,\underline{\alpha} \qquad (2.1.6)$$

Of course, multiplication of each of the vectors $\underline{\alpha}$ and $\underline{\beta}$ by an arbitrary -positive or negative- constant will produce equally valid weights, which maximize the correlation between u and v. The constraints (2.1.5) would then not be satified.

---

1) Letters in parentheses at the left and top margin identify sets of variables; letter on the bottom and at the right-hand margin denote the order of the matrices.

2.2 Categorical Analysis (Lancaster)

The results of section 2.1 can now be applied to a two-way contingency table of size p by q, where p is the number of levels of one categorized response variable and q is the number of states of the other response variable[2]. In order to scale the states of the response variables, dummy variables are introduced, as follows:

Let x_i be the dummy variable for "row" ($i = 1, 2, \ldots, p$) and y_j be that for "column" ($j = 1, 2, \ldots, q$), associated with each tally in the contingency table, there are p + q dummy variables.

$$\begin{cases} x_{ik} = 1, & \text{if the } k^{th} \text{ observation appear in row } i \\ \quad\;\; = 0, & \text{otherwise.} \\ y_{jk} = 1, & \text{if the } k^{th} \text{ observation appear in column } j \\ \quad\;\; = 0, & \text{otherwise.} \end{cases} \qquad (2.2.1)$$

where $k = 1, 2, \ldots, n$.

Therefore

$$\begin{cases} \sum_k x_{ik}^2 = n_{i.}\,, & \sum_k y_{jk}^2 = n_{.j} \\ \sum_k x_{ik} = n_{i.}\,, & \sum_k y_{jk} = n_{.j} \\ \sum\sum_k x_{ik}x_{hk} = \sum\sum_k y_{jk}y_{mk} = 0 \end{cases} \qquad (2.2.2)$$

since an observation cannot be in row i and row h or column j and column m at the same time.

---

2) Both expressions, "level" and "state" are customary; we will regard them as equivalent.

Let $s^{(x)}$ denotes the corrected sums of squares and products for the dummy variables $\underline{x}$; the i^{th} diagonal element is

$$s_{ii}^{(x)} = \Sigma x_{ik}^2 - \frac{(\Sigma x_{ik})^2}{n}$$

$$= n_{i.} - \frac{n_{i.}^2}{n} \qquad (2.2.3)$$

the (i,h) element is

$$s_{ih}^{(x)} = \Sigma x_{ik}x_{hk} - \frac{(\Sigma x_{ik})(\Sigma x_{hk})}{n}$$

$$= -\frac{n_{i.}\,n_{h.}}{n} \qquad (2.2.4)$$

If the k^{th} individual's tally occurs in row i and column j of the contingency table, there are thus (p + q) scores on the dummy variables, $\left[0,0,\ldots,0,\underset{(i)}{1},0,\ldots,0\right]$ on $\underline{x}$ and $\left[0,0,\ldots,0,\underset{(j)}{1},0,\ldots,0\right]$ on $\underline{y}$.

The matrix of sums of squares and products for the $\underline{x}$ variable is thus

$$E_{xx} = D_{n_{i.}} - \frac{\underline{n}_{i.}\underline{n}'_{i.}}{n} \quad (i=1,2,\ldots,p) \qquad (2.2.5)$$

where $D_{n_{i.}}$ is a diagonal matrix with typical element $n_{i.}$ (i=1,2,...,p) and $\underline{n}'_{i.}$ is the row vector $\left[n_{1.},n_{2.},\ldots,n_{p.}\right]$, the vector of row sums. For the response variable $\underline{y}$, the matrix of sums of squares and products is accordingly

$$E_{yy} = D_{n_{.j}} - \frac{\underline{n}_{.j}\underline{n}'_{.j}}{n} \quad (j=1,2,\ldots,q) \qquad (2.2.6)$$

where $D_{n_{.j}}$ is a diagonal matrix with typical element $n_{.j}$ $(j=1,2,\ldots,q)$ and $\underline{n}'_{.j} = [n_{.1}, n_{.2}, \ldots, n_{.q}]$, the vector of column sums.

Since $\Sigma x_{ik}y_{jk} = n_{ij}$ which is the cell frequency for cell (i,j) of the two-way contingency table, the corrected sums of products between x_i and y_j is

$$S_{ij}^{(xy)} = n_{ij} - \frac{n_{i.}\, n_{.j}}{n} \quad \begin{array}{l}(i=1,2,\ldots,p)\\(j=1,2,\ldots,q)\end{array} \tag{2.2.7}$$

in matrix form

$$E_{xy} = N - \frac{\underline{n}_{i.}\, \underline{n}'_{.j}}{n} \tag{2.2.8}$$

where N is the incidence matrix, and its typical element n_{ij} is the cell frequency in row i and column j of the contingency table.

Now we can perform the "Canonical Analysis" :

Since $E_{xx}^{(-1)} = D_{n_{i.}}^{-1}$ and $E_{yy}^{(-1)} = D_{n_{.j}}^{-1}$

$$E_{xx}^{(-1)}E_{xy}E_{yy}^{(-1)}E'_{xy} = D_{n_{i.}}^{-1}\left(N - \frac{\underline{n}_{i.}\underline{n}'_{.j}}{n}\right)D_{n_{.j}}^{-1}\left(N' - \frac{\underline{n}_{.j}\underline{n}'_{i.}}{n}\right)$$ [3)]

$$= \left(D_{n_{i.}}^{-1}N - \frac{\underline{j}\,\underline{n}'_{.j}}{n}\right)\left(D_{n_{.j}}^{-1}N' - \frac{\underline{j}\,\underline{n}'_{i.}}{n}\right) \tag{2.2.9}$$

where $\underline{j}' = [1,1,\ldots,1]$, thus $D_{n_{i.}}^{-1}\underline{n}_{i.} = \underline{j}$ and $D_{n_{.j}}^{-1}\underline{n}_{.j} = \underline{j}$.

---

3) $A^{(-1)}$ denotes a conditional inverse of A, i.e. any matrix for which $AA^{(-1)}A = A$ (Bargmann [1966], rule 9.1).

Hence

$$E_{xx}^{(-1)}E_{xy}E_{yy}^{(-1)}E'_{xy} = D_{n_{i.}}^{-1}\, N\, D_{n_{.j}}^{-1}\, N' - \frac{D_{n_{i.}}^{-1}\, N\underline{j}\, \underline{n}'_{i.}}{n}$$

$$= D_{n_{i.}}^{-1}\,(N\, D_{n_{.j}}^{-1}\, N' - \frac{\underline{n}_{i.}\, \underline{n}'_{i.}}{n}) \qquad (2.2.10)$$

since $N\underline{j} = \underline{n}_{i.}$

Denote the final matrix by Q^*; then the square of the maximum canonical correlation between the $\underline{x}$ set and the $\underline{y}$ set can be obtained by finding the largest characteristic root of Q^*, since

$$\rho^2 = Ch_{max}(Q^*) = \lambda \text{ , say.} \qquad (2.2.11)$$

We will determine the λ and the associated characteristic vector $\underline{a}$, so that

$$Q^*\, \underline{a} = \lambda\, \underline{a} \qquad (2.2.12)$$

Let $W = (N\, D_{n_{.j}}^{-1}\, N' - \frac{\underline{n}_{i.}\, \underline{n}'_{i.}}{n})$ (2.2.13)

then $D_{n_{i.}}^{-1}\, W\, \underline{g} = \lambda\, \underline{g}$ (2.2.14)

where $\underline{g}$ is a characteristic vector.

Let $\underline{g} = D_{n_{i.}}^{-\frac{1}{2}}\, \underline{a}$ and (2.2.15)

premultiply $D_{n_{i.}}^{-1}\, W\underline{g} = \lambda\, \underline{g}$ by $D_{n_{i.}}^{\frac{1}{2}}$; then

$$D_{n_{i.}}^{-\frac{1}{2}}\, W\, \underline{g} = \lambda\, D_{n_{i.}}^{\frac{1}{2}}\, \underline{g} \qquad (2.2.16)$$

therefore

$$D_{n_{i.}}^{-\frac{1}{2}} W D_{n_{i.}}^{-\frac{1}{2}} \underline{a} = \lambda \underline{a} \tag{2.2.17}$$

and since Ch(ABC) = Ch(CAB), hence

$$\rho^2 = Ch_{max}(Q^*) = Ch_{max}(D_{n_{i.}}^{-1} W) = Ch_{max}(D_{n_{i.}}^{-\frac{1}{2}} W D_{n_{i.}}^{-\frac{1}{2}}) \tag{2.2.18}$$

and, denoting by Q the symmetric matrix in (2.2.17) then

$$Q \underline{a} = \lambda \underline{a} ,$$

so $$\underline{b} = - E_{yy}^{(-1)} E_{xy}' \underline{a} = -D_{n_{.j}}^{-1} \left(N' - \frac{\underline{n}_{.j} \underline{n}_{i.}'}{n}\right) \underline{a} \tag{2.2.19}$$

are the weights that maximize $|corr(u,v)|$ where

$$u = \underline{a}'\underline{x} \quad \text{and} \quad v = \underline{b}'\underline{y} . \tag{2.2.20}$$

Now, recall that $\underline{x}$ and $\underline{y}$ are (0, 1) variables; let $\underline{x}_k'$ denote the $\underline{x}$-vector of the k^{th} subject and $\underline{y}_k'$ denote his $\underline{y}$-vector. If he was in state i in variable x (rows) and in state j in variable y (columns), $u_k = \underline{a}'\underline{x}_k = a_i$ and $v_k = \underline{b}'\underline{y}_k = b_j$, thus a_i is the scaled response for state i in x , and b_j is the scaled response for state j in y.

CHAPTER III

MEASUREMENT OF INTERNAL DEPENDENCE IN A SINGLE SET OF VARIABLES

3.1 Generalization of the Canonical Correlation Concept

Before we can generalize the bivariate results of CHAPTER II, we must discuss the generalization of the canonical correlation concept.

Let $\underline{x}$ be a given single set of variables with p variables in the set, and let

$$P = \begin{bmatrix} 1 & \rho_{12} & \rho_{13} & \cdots & \rho_{1p} \\ \rho_{12} & 1 & \rho_{23} & \cdots & \rho_{2p} \\ \rho_{13} & \rho_{23} & 1 & \cdots & \rho_{3p} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \rho_{1p} & \rho_{2p} & {}_{3p} & \cdots & 1 \end{bmatrix} \qquad (3.1.1)$$

be their correlation matrix. Let E be a matrix of corrected sums of squares and products [1] for $\underline{x}$ and let Σ be the covariance matrix. Let $D_{e_{ii}}$ be a diagonal matrix with diagonal elements of E in the principal diagonal and zero elsewhere. The sample correlations are then

$$r_{ij} = \frac{e_{ij}}{\sqrt{e_{ii}\, e_{jj}}} \quad \text{and} \quad r_{ii} = 1 \qquad (3.1.2)$$

---

1) Or a matrix of S.S. and S.P. error if the sample came from some same analysis of variance design

Since we want to test internal dependence, a null hypothesis is H_o: P = I; this is equivalent to saying that the covariance matrix is a diagonal matrix i.e. $H_o: \sum = D$ where the principal diagonal elements are σ_{ii} (arbitrary) and off-diagonal elements are zero. Then, as is well known, the likelihood-ratio test statistic is $|R|$, the determinant of the sample correlation matrix.

To compare with available table, one uses $-m \ln|R|$ as a test statistic distributed under null hypothesis as a series of χ^2 -variables (Wald-Brookner [1941], Morrison [1967]).

Thus $|R|$ is the likelihood-ratio statistic for the test of internal independence. Since all diagonal elements are unity, it has a maximum value of 1 (which occurs when R = I) and a minimum value of 0 (since $|R|$ is, of course, positive-definite or positive-semidefinite). We will demonstrate in the next section that maximization (or minimization) of parametric functions analogous to likelihood-ratio statistics is equivalent to obtaining maximum likelihood estimates. Hence, we will obtain weights in k-dimensional categorical scaling by minimizing the determinant of the resulting correlation matrix.

3.2 Justification of Minimum-Determinant Criterion

Among many different index numbers for measuring the multiple dependence between sets of variables, we choose

Steel's "Minimum Correlation Determinant", with some generalized transformation on each set for the following reason:

There is, in all known situations, a relationship between estimation by minimizing a likelihood-ratio statistic (LR) and maximum likelihood estimation (MLE). We will demonstrate this relationship below for several simple examples. We seek a parametric function, and maximum likelihood estimators have the invariance property, i.e. under certain conditions of uniqueness, if t is the maximum likelihood estimator of θ then $f(t)$ is the maximum likelihood estimator of $f(\theta)$. It is this relationship between a function of maximum likelihood estimators and the corresponding function of parameters which is used to obtain the parametric function. In the present instance we cannot formulate a parametric model for which the "Minimum Correlation Determinant" estimates would be maximum likelihood estimates. This is not necessarily a hindrance to its use, for a similar lack exists in Factor Analysis, and even in multivariate analysis of variance, where the rather artificial non-centrality parameters must be introduced before the problem can be stated as one of maximum likelihood estimation (Bargmann [1969]).

Estimation by "Minimum Likelihood-Ratio" is, of course similar to "Minimum Chi-square Estimation" (e.g. Cramér [1946]). In either case, parameters are estimated in such a way that a test statistic ("Goodness-of-fit" statistic) is

minimized. It is, of course, well known that the "modified" minimum Chi-square Statistic (Cramér, ibid), i.e. the goodness-of-fit statistic with E_i omitted in the denominator, leads to maximum likelihood estimates. The choice of the minimum determinant procedure is thus based on a conjecture which has not been proven (except in the trivial cases where the likelihood-ratio can attain the value 1) but for which no counterexample is known. The conjecture is (as in the case of modified minimum Chi-square) that estimates obtained by minimizing likelihood-ratio statistics are maximum likelihood estimates of parameters for some model that fits the sample most closely.

Example:

Estimation of a common variance.

a) For the Simple Univariate Case:

Given a sample of size n_1 from $N(\mu_1, \sigma_1^2)$ and a sample of size n_2 from $N(\mu_2, \sigma_2^2)$:

In Ω we have

$$\ln L(\hat{\Omega}) = -\frac{n}{2}\ln(2\Pi) - \frac{n_1}{2}\ln\hat{\sigma}_1^2 - \frac{n_2}{2}\ln\hat{\sigma}_2^2 - \frac{n}{2} \tag{3.2.1}$$

where $\hat{\mu}_i = \dfrac{\Sigma\, y_{ij}}{n_i}$, $\hat{\sigma}_i^2 = \dfrac{\Sigma\,(y_{ij} - \mu_i)^2}{n_i}$.

In ω , $\sigma_1^2 = \sigma_2^2 = \sigma^2$. This common σ^2 is to be estimated in such a way that the resulting likelihood-ratio statistic becomes a maximum. The logarithm of the likelihood

function is

$$\ln L(\omega|\mu_1 = \hat{\mu}_1, \mu_2 = \hat{\mu}_2)$$

$$= -\frac{n}{2}\ \ln(2\pi) - \frac{n}{2}\ln\sigma^2 - \frac{1}{2\sigma^2}(n_1\ \hat{\sigma}_1^2 + n_2\ \hat{\sigma}_2^2) \qquad (3.2.2)$$

hence, as a function of σ^2

$$\ln\lambda = -\frac{n}{2}\ln\sigma^2 - \frac{1}{2\sigma^2}(n_1\ \hat{\sigma}_1^2 + n_2\ \hat{\sigma}_2^2)$$

$$+\frac{n_1}{2}\ln\hat{\sigma}_1^2 + \frac{n_2}{2}\ln\hat{\sigma}_2^2 + \frac{n}{2} \qquad (3.2.3)$$

$$\frac{\partial\ln\lambda}{\partial(\sigma^2)^{-1}} = \frac{n}{2}\sigma^2 - \frac{1}{2}(n_1\ \hat{\sigma}_1^2 + n_2\ \hat{\sigma}_2^2) \qquad (3.2.4)$$

Trivially, $$\hat{\sigma}^2 = \frac{n_1\ \hat{\sigma}_1^2 + n_2\ \hat{\sigma}_2^2}{n} \qquad (3.2.5)$$

b) In the Multivariate Case:

We wish to test $H_o: \Sigma_1 = \Sigma_2$.

In ω , the likelihood-ratio statistic will be expressed as a function of the common Σ .

Let be the matrix of sums of squares and products for error in sample No. 1, E_2 the same for the sample No.2.

Let $\hat{\Sigma}_1 = \frac{1}{n_1}E_1$, $\hat{\Sigma}_2 = \frac{1}{n_2}E_2$, then

$$\ln L(\hat{\Omega}) = -\frac{np}{2}\ln(2\pi) - \frac{n_1}{2}\ln(\frac{1}{n_1}E_1) - \frac{n_2}{2}\ln(\frac{1}{n_2}E_2)$$

$$-\frac{np}{2} \qquad (3.2.6)$$

In ω , $\Sigma_1 = \Sigma_2 = \Sigma$ and the ln L(ω) as a function of Σ , is

$$-\frac{np}{2}\ln(2\pi) - \frac{n_1}{2}\ln|\Sigma| - \frac{n_2}{2}\ln|\Sigma| - \frac{1}{2}\operatorname{tr}\Sigma^{-1}(E_1 + E_2) \tag{3.2.7}$$

thus

$$\ln\lambda = \frac{n}{2}\ln\Sigma^{-1} - \frac{1}{2}\operatorname{tr}\Sigma^{-1}(E_1 + E_2) + \frac{n_1}{2}\ln(\frac{1}{n_1}E_1) + \frac{n_2}{2}\ln(\frac{1}{n_2}E_2) + \frac{np}{2} \tag{3.2.8}$$

$$\frac{\partial\ln\lambda}{\partial\Sigma^{-1}} = \frac{n}{2}\Sigma - \frac{1}{2}(E_1 + E_2) \tag{3.2.9}$$

is that value which maximizes the likelihood-ratio statistic $\ln\lambda$; equating $\frac{\partial\ln\lambda}{\partial\Sigma^{-1}}$ to zero, we obtain $\hat{\Sigma} = \frac{E_1 + E_2}{n}$, the well known pooled maximum likelihood estimate.

c) <u>In the Uniform Distribution Case</u>:

We discuss the hypothesis H_0: $\theta = \theta_0$ in $y = [0, \theta]$. The likelihood function is

$$\begin{cases} L(y) = (\frac{1}{\theta})^n & \text{if } y_{max} \quad \theta \\ \quad = 0 & \text{otherwise.} \end{cases} \tag{3.2.10}$$

Hence in Ω , $\theta = y_{max}$, since θ cannot be less than y_{max}.

$$L(\hat{\Omega}) = (\frac{1}{y_{max}})^n \tag{3.2.11}$$

$$\lambda = (\frac{y_{max}}{\theta_0})^n \tag{3.2.12}$$

Since θ_o must be greater than y_{max}, the θ_o which maximizes λ is y_{max}, which is also the maximum likelihood estimate of θ.

For a more complicated relationship (e.g. estimation in Factor Analysis) see Howe [1955] and Bargmann [1957].

Because of the striking connection between minimum likelihood-ratio and maximum likelihood estimation, we will now attack the problem of canonical weights in k sets of variables by choosing the weights in such a way that the determinant of the resulting correlation matrix is a minimum (since a large determinant for the correlation matrix indicates approach to independence).

Let $\underline{a}_i$ be a set of weights for the i^{th} set, and let

$$u_i = \underline{a}_i' \underline{y}_i \text{ then } \operatorname{corr}(u_i,u_j) = \frac{\underline{a}_i' R_{ij} \underline{a}_j}{h_i h_j} \text{ 2)} \qquad (3.2.13)$$

where $h_i = \sqrt{\underline{a}_i' \underline{a}_i}$.

Since, for the u-set, a test of independence would employ the determinant of the resulting matrix of sample correlation, we will estimate the weights $\underline{a}_i'$ in such a way that $|P|$ is a minimum, where ρ_{ij}, the elements of P, are

$$\rho_{ij} = \operatorname{corr}(u_i,u_j) \qquad (3.2.14)$$

and hence functions of the unknown $\underline{a}_i$.

---

2) We use R for the large matrix of correlation between the y's, even though the elements are parameters (ρ's), so that we can use P for the small matrix of correlation between the canonical variables.

This is Steel's approach to obtaining a single measure of dependence for k sets of variates. It has obvious relevance to the problem of categorical scaling, where the $\underline{y}_i$-sets will be dummy variables, and hence that $\underline{a}_i$ which minimizes the determinant will have the scale values for the states of the i^{th} categorical variables.

CHAPTER IV

MULTI-DIMENSIONAL CATEGORICAL SCALING

4.1 Conversion of Contingency Tables to Reduced Correlation Matrix

4.1.1 Description of Algorithm

The information required for multi-dimensional categorical scaling is contained in the k-way contingency table. An analysis which employs moments only up to the second order derives its information from every possible two-dimensional marginal of these tables. This restriction is, necessarily, the same as that in any other instance where the central limit theorem is employed (e.g. sign-tests, goodness-of-fit, Chi-square tests). In obtaining the elements of the super-correlation matrix we thus proceed, for each pair of categorical variables, directly as in section 2.2 (formula 2.2.5, 2.2.6 and 2.2.8). For example, for four sets we construct a matrix

$$\begin{bmatrix} E_{11} & E_{12} & E_{13} & E_{14} \\ E'_{12} & E_{22} & E_{23} & E_{24} \\ E'_{13} & E'_{23} & E_{33} & E_{34} \\ E'_{14} & E'_{24} & E'_{34} & E_{44} \end{bmatrix} \qquad (4.1.1.1)$$

where E_{11} has elements

$$e_{ii}^{(1,1)} = n_{i\ldots} - \frac{n_{i\ldots}^2}{n} \qquad (4.1.1.2)$$

$$e_{ij}^{(1,1)} = e_{ji}^{(1,1)} = -\frac{n_{i\ldots}n_{j\ldots}}{n} \qquad (4.1.1.3)$$

E_{12} has elements

$$e_{ij}^{(1,2)} = n_{ij..} - \frac{n_{i\ldots}n_{.j..}}{n} \qquad (4.1.1.4)$$

... etc

E_{34} has elements

$$e_{ij}^{(3,4)} = n_{..ij} - \frac{n_{..i.}n_{\ldots j}}{n} \qquad (4.1.1.5)$$

and finally, E_{44}

$$e_{ii}^{(4,4)} = n_{\ldots i} - \frac{n^2_{\ldots i}}{n} \qquad (4.1.1.6)$$

and $$e_{ij}^{(4,4)} = -\frac{n_{\ldots i}n_{\ldots j}}{n} \qquad (4.1.1.7)$$

For p categorical variables, there will thus be a super-matrix of corrected sums of squares and products, hence

$$\begin{array}{cc} \begin{bmatrix} E_{11} & E_{12} & \cdots & E_{1p} \\ E'_{12} & E_{22} & \cdots & E_{2p} \\ \cdot & \cdot & \cdots & \cdot \\ E'_{1p} & E'_{2p} & \cdots & E_{pp} \end{bmatrix} & \begin{array}{c} (\ell_1) \\ (\ell_2) \\ \\ (\ell_p) \end{array} \\ \begin{array}{cccc} (\ell_1) & (\ell_2) & \cdots & (\ell_p) \end{array} & \end{array} \qquad (4.1.1.8)$$

where E_{ij} is of order $(\ell_i \times \ell_j)$, and ℓ_i denotes the number of states or levels of categorical variable i.

For the maximum likelihood estimation of scale values (weights) it would now be necessary to reduce the E matrix

to a matrix of correlations. Steel [1949] , however, suggested a further reduction to a correlation super-matrix in which the diagonal submatrices are identity matrices. We note further that the E matrix has rank $\ell_1 + \ell_2 + \ldots + \ell_p - p$; hence reduction of E to a normalized R of that smaller order would be desirable.

Since E_{ii} is Gramian of rank $(\ell_i - 1)$, there exists real-valued, rectangular matrices T_i (order $\ell_i \times (\ell_i - 1)$) such that $T_i T_i' = E_{ii}$; among the infinitely many T_i's satifying this relation we prefer to use that which can be obtained from the Gauss-Doolittle algorithm (see formula 11.10 and 9.14 of Bargmann [1966]), mainly because the same algorithm also produces a conditional inverse from the left, i.e. a matrix $T_i^{(-1)}$ such that $T_i^{(-1)} T_i = I \; ((\ell_i - 1) X (\ell_i - 1))$. If we now premultiply E by the matrix

$$T^{(-1)} = \begin{array}{cc} \begin{bmatrix} T_1^{(-1)} & 0 & \ldots & 0 \\ 0 & T_2^{(-1)} & \ldots & 0 \\ \cdot & \cdot & \ldots & \cdot \\ 0 & 0 & \ldots & T_p^{(-1)} \end{bmatrix} & \begin{array}{c} (\ell_1 - 1) \\ (\ell_2 - 1) \\ \\ (\ell_p - 1) \end{array} \\ \begin{array}{cccc} (\ell_1) & (\ell_2) & \quad & (\ell_p) \end{array} & \end{array} \qquad (4.1.1.9)$$

and postmultiply by $T^{(-1)'}$, the transpose of the above matrix, we will obtain a matrix

$$R = \begin{array}{cc} \begin{bmatrix} I & R_{12} & \cdots & R_{1p} \\ R'_{12} & I & \cdots & R_{2p} \\ \cdot & \cdot & \cdots & \cdot \\ R'_{1p} & R'_{2p} & \cdots & I \end{bmatrix} & \begin{array}{c} (\ell_1-1) \\ (\ell_2-1) \\ \\ (\ell_p-1) \end{array} \\ \begin{array}{cccc} (\ell_1-1) & (\ell_2-1) & & (\ell_p-1) \end{array} & \end{array} \qquad (4.1.1.10)$$

which is non-singular, and has a reduced form which makes further calculation easier.

4.1.2 Description of the Computer Program for C-E-R

This is the computer program for the contingency table to E matrix to R matrix, in brief: C-E-R. A listing of this computer program is given in Appendix A2. The layout for INPUT is in Appendix A1.

The computer program is written in FORTRAN IV and has been used on an IBM 360/65 (and also, with several overlays, on an IBM 1130). It uses some of the routine from the IBM Scientific Subroutine Package (SSP).

This computer program is designed to read a set of records with the necessary parameters (in FORMAT 5I2) and a multiple dimension contingency table (in FORMAT 4I2,5I4) and up to five sets, each set would have up to five levels.

The super-matrix index is designed as follow; where K is an index stepped up in accordance with IBM SSP storage mode 1 packing.

1,1 K=1	1,2 K=2	1,3 K=4	1,4 K=7	1,5 K=11
	2,2 K=3	2,3 K=5	2,4 K=8	2,5 K=12
		3,3 K=6	3,4 K=9	3,5 K=13
			4,4 K=10	4,5 K=14
				5,5 K=15

(4.1.2.1)

The computer program proceeds as follows:

(1) READ in the parameters:

NVAR - No. of sets (response variables).

ND1 - No. of levels in the first set.

ND2 - No. of levels in the second set.

ND3 - No. of levels in the third set.

ND4 - No. of levels in the fourth set.

ND5 - No. of levels in the fifth set.

READ in the multiple dimension contingency table for cell frequencies.

(2) Using a FUNCTION subprogram LT, this program assigns all entries of all contingency tables into a single dimension array AR.

(3) Pick up the limit of K i.e. the number of submatrices contained in the upper triangular part of E.

If NVAR = 2 then KMAX= 3.

If NVAR = 3 then KMAX = 6.

If NVAR = 4 then KMAX = 10.

If NVAR = 5 then KMAX = 15.

(4) The program now proceeds to (5) or (6) depending on whether submatrices are in the diagonal or off-diagonal portions of E..

(5) If K = 1, 3, 6, 10, 15 then call the subroutines to construct the E_{ii} matrix for i =,1, 2, 3, 4, 5.

(6) If K $\neq$ 1, 3, 6, 10, 15 then call the subroutine to construct E_{ij} for $i \neq j$, the off-diagonal submatrices in the super-matrix E.

(7) Compare the index with the corresponding NVAR designated KMAX value. If they are not equal then go back to loop to finish the construction of E matrix otherwise go to next step.

(8) After all E_{ii} matrices have been constructed, proceed to the calculation of the T conditional inverses for all the diagonal submatrices in E.

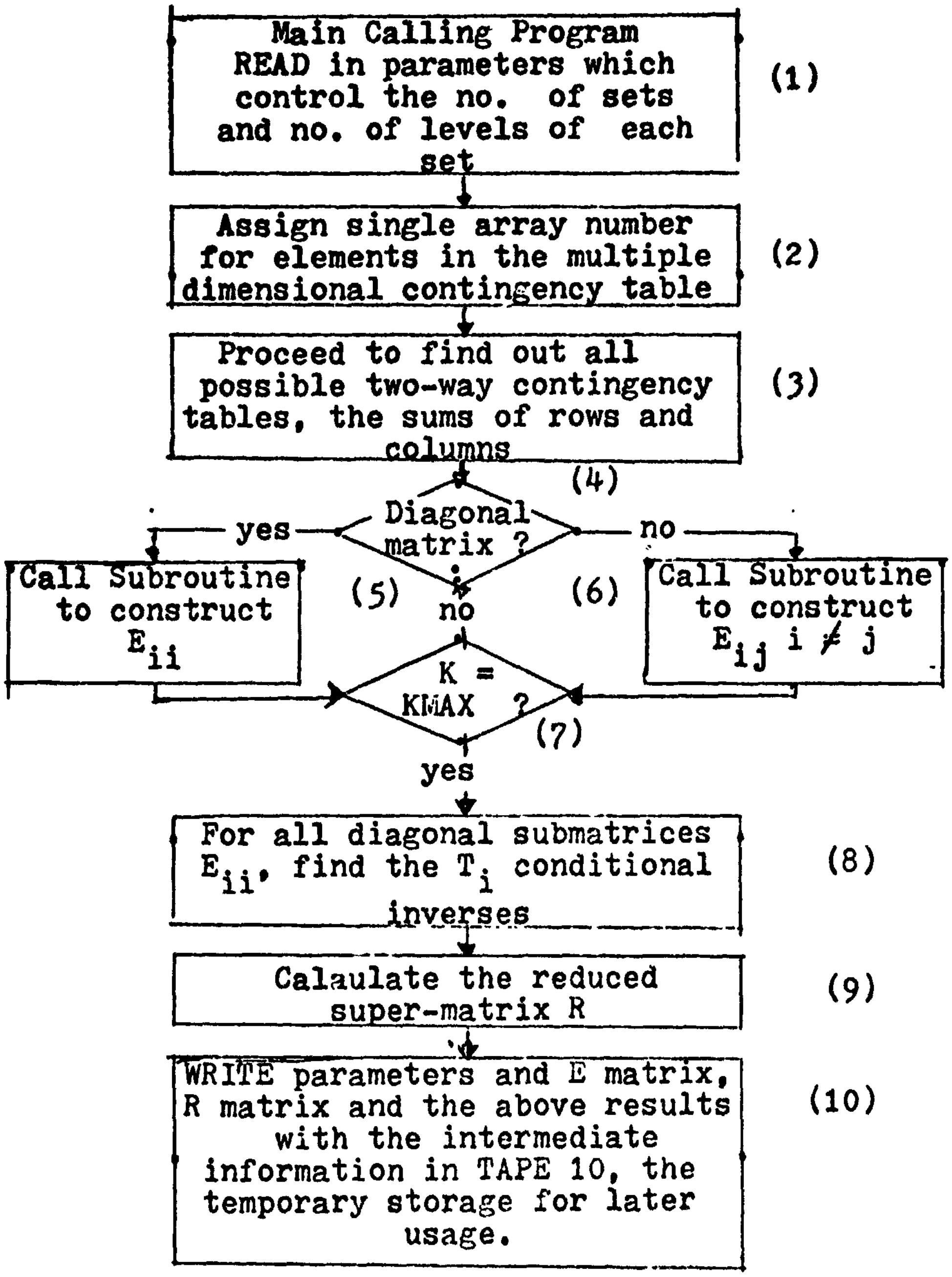
(9) Calculate $R_{ij} = T_i^{(-1)} E_{ij} T_j^{(-1)'}$.

(10) <u>OUTPUT</u> the parameters and the multiple dimensional contingency tables as in (1), the E and R matrices, also the T conditional inverses. The above results along with marginal totals for each set and the grand total are written in TAPE 10, the temporary tape storage for further usage. All these intermediate results will be used later to express the scale values in terms of the original states.

The card layout is described in Appendix A1, the reason for using Response Variable (2) as the last subscripted variable is that when k-way contingency tables are recorded, it is customary to record them as sets of two-way tables with the third, fourth, etc dimensions fixed. Each two-way table then has the first dimension as rows and the second dimension as columns. To avoid key punch errors it is advisable to punch the cards for such pairwise contingency tables so that there is one card per row. Thus, if the k-dimensional array is expanded into a one-dimensional string, the levels of Response (1) should vary fastest; however, the levels of Response (2) are columns of a two-way contingency table which, if key-punched row-wise, would occur on the same punched card.

<u>Computer Program C-E-R</u>

<u>Flow-Chart</u>



An example of INPUT and OUTPUT data of the C-E-R computer program as follow:

(1) INPUT data:

a. NVAR = 3, ND1 = 3, ND2 = 3, ND3 = 3, ND4 = 1,[1)] ND5 = 1

b. Contingency table:

1	1	1	1	30	70	30
2	1	1	1	10	50	40
3	1	1	1	75	20	5
1	2	1	1	37	28	10
2	2	1	1	15	25	30
3	2	1	1	27	0	23
1	3	1	1	53	5	7
2	3	1	1	35	25	20
3	3	1	1	30	0	20

(2) OUTPUT information:

Direct print-out from the computer is on page 42 - 46.

THE CONTINGENCY TABLE INPUT DATA

THE NUMBER OF SETS NSETS) = 3

THE NUMBER OF RESPONSES OF FIRST LEVEL (ND1) = 3

THE NUMBER OF RESPONSES OF SECOND LEVEL (ND2) = 3

THE NUMBER OF RESPONSES OF THIRD LEVEL (ND3) = 3

THE NUMBER OF RESPONSES OF FOURTH LEVEL (ND4) = 1 [1)]

THE NUMBER OF RESPONSES OF FIFTH LEVEL (ND5) = 1 [1)]

CONTINGENCY TABLE

LEVEL (1)	LEVEL (3)	LEVEL (4)	LEVEL (5)	LEVEL (2) (1... 3)		
1	1	1	1	30	70	30
2	1	1	1	10	50	40
3	1	1	1	75	20	5
1	2	1	1	37	28	10
2	2	1	1	15	25	30
3	2	1	1	27	0	23
1	3	1	1	53	5	7
2	3	1	1	35	25	20
3	3	1	1	30	0	20

1) LEVEL = 1 indicates that there is no fourth and fifth responses.

THE E 1 1 MATRIX

0.168750E 03	-0.937500E 02	-0.750000E 02
-0.937500E 02	0.163194E 03	-0.694444E 02
-0.750000E 02	-0.694444E 02	0.144444E 03

THE E 1 2 MATRIX

0.300000E 01	0.193750E 02	-0.223750E 02
-0.483333E 02	0.225694E 02	0.257638E 02
0.453333E 02	-0.419444E 02	-0.338888E 01

THE E 2 2 MATRIX

0.176800E 03	-0.966333E 02	-0.801666E 02
-0.966333E 02	0.153931E 03	-0.572986E 02
-0.801666E 02	-0.572986E 02	0.137465E 03

THE E 1 3 MATRIX

0.625000E 01	0.187500E 01	-0.812500E 01
-0.145833E 02	0.229166E 01	0.122916E 02
0.833333E 01	-0.416666E 01	-0.416666E 01

THE E 2 3 MATRIX

-0.280000E 02	-0.550000E 01	0.335000E 02
0.377916E 02	-0.739583E 01	-0.303958E 02
-0.979166E 01	0.128958E 02	-0.310416E 01

THE E 3 3 MATRIX

0.178750R 03	-0.893750E 02	-0.893750E 02
-0.893750E 02	0.142187E 03	-0.528125E 02
-0.893750E 02	-0.528125E 02	0.142187E 03

INFORMATION STORED IN DISK

FILE NAME IS $$$$$

NUMBER OF VARIABLES = 6

MAXIMUM NUMBER OF ITERATIONS = 150

MINIMUM VALUE OF THE FUNCTION = 0.000000E 00

PERMISSIBLE ERROR DURING ITERATION = 0.100000E-02

NUMBER OF SETS (NSETS) = 3

NUMBER OF ROWS (1,1) = 2

NUMBER OF ROWS (2,2) = 2

NUMBER OF ROWS (3,3) = 2

NUMBER OF ROWS (4,4) = 0

NUMBER OF ROWS (5,5) = 0

WRITE IN WEIGHTS [2)]

0.500000E 00	0.500000E 00	0.500000E 00
0.500000E 00	0.500000E 00	0.500000E 00

THE MARGINAL TOTAL FOR SET 1

270	250	200

THE MARGINAL TOTAL FOR SET 2

312	223	185

THE MARGINAL TOTAL FOR SET 3

330	195	195

THE GRAND TOTAL = 720

---

2) Dummy numbers at this stage.

THE T CONDITIONAL INVERSE

0.769800E-01	0.000000E 00	0.000000E 00
0.527046E-01	0.948683E-01	0.000000E 00

THE T CONDITIONAL INVERSE

0.752071E-01	0.000000E 00	0.000000E 00
0.543546E-01	0.994470E-01	0.000000E 00

THE T CONDITIONAL INVERSE

0.747957E-01	0.000000E 00	0.000000E 00
0.506369E-01	0.101273E 00	0.000000E 00

THE R11 MATRIX

0.100000E 01	0.000000E 00
0.000000E 00	0.100000E 01

THE R12 MATRIX

0.173683E-01	0.160876E 00
-0.332955E 00	0.738409E-01

THE R22 MATRIX

0.100000E 01	0.000000E 00
0.000000E 00	0.100000E 01

THE R13 MATRIX

0.359861E-01	0.389803E-01
-0.788416E-01	-0.213504E-01

THE R23 MATRIX

-0.157504E 00	-0.148522E 00
0.167268E 00	0.847920E-02

THE R33 MATRIX

0.100000E 01	0.000000E 00
0.000000E 00	0.100000E 01

4.2 Initial Estimate and Approximate Scale Values

4.2.1 Description of Algorithm

After the reduced super-matrix R has been constructed, the minimization of a given function (the log determinant of the resulting correlation matrix P) is needed to obtain the categorical scales for each set; we must first find some initial values for categorical scales to start the minimization.

The Fletcher and Powell Descent Method [1963] is used for the minimization. Initial values must be chosen very carefully, so that the numerical analysis converges in a reasonable number of steps (or, for that matter, converges at all). The following methods are available:

4.2.1.1 Arbitrary Determination of the Initial Values

The initial values for categorical scales can be determined arbitrarily. We may start with all 1's or all 0.5's or any other values which we may conveniently think of. It is very unusual for this method to converge.

4.2.1.2 Average Canonical Scales

Starting from the off-diagonal matrices of the super-matrix R, obtain

$$Q_i = R_{ij} R'_{ij} \qquad (4.3.1.2.1)$$

where R'_{ij} is the transpose of R_{ij}, and the largest characteristic root $\lambda_i = Ch_{max}(Q_i)$, with associated

vector $\underline{u}_i$. Note that $\underline{u}_i$ is not the same when response (i) is compared with response (k) $\neq$ (j). Thus, for each response variable we have (p - 1) different characteristic vectors. Of course the characteristic root - characteristic vector analysis needs to be done for p(p - 1)/2 matrices only since, if

$$R_{ij} R'_{ij} \underline{u}_i = \lambda \underline{u}_i \tag{4.2.1.2.2}$$

then $R'_{ij} R_{ij}$ has the same largest characteristic root λ, and the associated characteristic vector, say $\underline{u}^*_j$, equals $R'_{ij} \underline{u}_i$.

As a starting value we may average the (p - 1) $\underline{u}_i$ vectors obtained for each pairing of the i^{th} response with the others. As in every characteristic root - characteristic vector analysis, the length of the $\underline{u}_i$ are indeterminate. In the above-mentioned method they are all taken to be unit length. Thus, if

$$R_{i1} R'_{i1} \underline{u}_{i(1)} = \lambda_1 \underline{u}_{i(1)} \tag{4.2.1.2.3}$$

and $\underline{u}'_{i(1)} \underline{u}_{i(1)} = 1$,

then $$\underline{v}_i = \frac{\sum_{j \neq i} u_{i(j)}}{(p - 1)} \tag{4.2.1.2.4}$$

$\underline{v}_i$ is the initial vector of weights for response i. The weakness of this averaging procedure is that the same weights are applied to each characteristic vector, regardless of whether response i has a low or high correlation with the other responses with which it is paired.

4.2.1.3 <u>This leads to a further improvement of the starting value</u>

As in 4.2.1.2, we obtain all (p - 1) characteristic vectors for each response i; our initial estimate now is a weighted average, using the largest characteristic root for each pairing as weights. Thus, if $Ch(R_{i1}R'_{i1})=\lambda_1$ and the corresponding characteristic vector is $\underline{u}_{i(1)}$ then

$$\underline{v}_i = \frac{\sum_{j\neq i}\lambda_j \, \underline{u}_{i(j)}}{\sum_{j\neq i}\lambda_j} \qquad (4.2.1.3.1)$$

is the initial vector of weights for response i. This method works very well (see section 6.6); there is, however, a problem regarding the signs of $\underline{u}_{i(j)}$, since it is unknown whether, say, (+ - + +) or (- + - -) is the "proper" sign (possibility of "negative canonical correlation"). For the weights which are appreciably different from zero we used the sign pattern obtained for the characteristic vector associated with the largest of the (p - 1) maximum characteristic root, so as to define the largest correlation to be positive.

4.2.1.4 <u>Multiple Regreesion Approach</u>

If the (ℓ_i - 1) dummy variables in response i are considered as concomitant variables, and the (ℓ_j - 1) dummy variables in response j are considered as (ℓ_j - 1) predictands (response variables), then the matrix R_{ij} would be a matrix of regression weights estimates,

since $R_{ii} = I$ and $R_{ii}^{-1} R_{ij} = B$, the regression weights estimates. From this analogy, another starting value can be obtained. To determine the weights to be applied to the (p - 1) $\underline{u}_i$ vectors (characteristic vectors) we set up a (p - 1)x(p - 1) matrix A whose elements are the canonical correlations (positive square-roots of largest characteristic roots of $R_{jk} R_{jk}'$) between the response sets other than i. As a right-hand side for the regression-like equation we use the canonical correlations of response set i versus the other (p - 1) response sets. The multiple regression weights obtained as solutions from this system of equations are used as weights for the averaging of the $\underline{u}_{i(j)}$ (in place of the λ_j in 4.2.1.3.1).

4.2.1.5 Canonical-multiple Correlation Approach

For each response set partition the super-matrix R into two parts

$$\begin{array}{cc} \begin{bmatrix} I & R^*_{(i)} \\ R^{*\prime}_{(i)} & R^*_{(jj)} \end{bmatrix} & \begin{array}{c} (\ell_i - 1) \\ (\ell) \end{array} \\ \begin{array}{cc} (\ell_i - 1) & (\ell) \end{array} & \end{array} \qquad (4.2.1.5.1)$$

where $(\ell) = \Sigma \ell_j - p + 1$

$$R^*_{(i)} = [\; R_{i1} \;\; R_{i2} \;\; \ldots \;\; R_{ip} \;] \qquad (4.2.1.5.2)$$

(except R_{ii})

and

$$R^*_{(jj)} = \begin{bmatrix} I & R_{12} & \cdots & R_{1p} \\ R'_{12} & I & \cdots & R_{2p} \\ \cdot & \cdot & \cdots & \cdot \\ R'_{1p} & R'_{2p} & \cdots & I \end{bmatrix} \qquad (4.2.1.5.3)$$

(except the i^{th} pseudo-row and pseudo-column).

Thus obtain, for each response set, a single set of weights $\underline{u}_i$ which represents the canonical weights on the i^{th} response set when correlated with all other sets combined.

4.2.2 <u>Description of the Computer Program for ICW</u>

This is the computer program for obtaining the initial weights, in brief: ICW. The listing of the computer program is in Appendix B. The IBM Scientific Subroutine Package program used is <u>EIGEN</u> which obtains characteristic roots and vectors of real symmetric matrices using the <u>Jacobi Method</u> (see <u>System/360-SSP-360A-CM-03X-Version III, Programmer's Manual</u>, 164).

(1) The main calling program <u>READ</u> the information from TAPE 10.

i. The 1st record: which will not be used in this program.

ii. The 2nd record: NSET - no. of sets.

NRST(I),I=1,NSET - the size of R_{ij} submatrices of R.

iii. The 3rd record: X(I),I=1,N - dummy numbers of weights.

iv. The 4th, 5th and 6th are not used in this program.

(2) Initial LIMIT = 1.

(3) Subroutine REAIN - Read the 7th read from TAPE 10, the super-matrix R which is the reduced matrix from the program C-E-R.

(4) If LIMIT = 1, proceed to (5); if LIMIT = 2 proceed to (8).

(5) Subroutine RIOUT - It will output the information on the 2nd and 7th records, i.e. no. of sets, sizes of submatrices and the super-matrix R.

(6) Provides the working storage for the R^* set, the Subroutine EMPTY will zero out the working storage.

(7) Subroutine RS1 - Calculates the $R^* = R R'$ where R' is the transpose of R, then proceeds to (11).

(8) Subroutine ROUT - Output the R' matrix.

(9) Same as (6).

(10) Subroutine RS2 - Calculates the $R^* = R'R$ matrix then proceeds to (11).

(11) Packed the R^* matrix into a single-dimensioned array with storage mode 1, in order to use the SSP EIGEN.

(12) Subroutine EIGEN - Find the largest characteristic root and the associated characteristic vector.

(13) Subroutine SECOD - If LIMIT =1, the roots and vectors

and the square root of the characteristic roots will be padked into appropriate single-dimensioned variables for later usage. If LIMIT = 2, the Subroutine will do the same thing except the single-dimensioned variables will be different from those when LIMIT = 1.

(14) If NSET is greater than 2, proceeds to (15), otherwise proceeds to (8).

(15) Subroutine RLOAD - Reload the other R^* matrix then set IGO value. If IGO = 2, proceeds to (11). If IGO = 1, proceeds to (16).

(16) If LIMIT $\neq$ 1 then proceeds to (18), if LIMIT = 1, proceeds to (17).

(17) Set LIMIT = 2 then proceeds to (4).

(18) Subroutine FINAL - It will sum and average the characteristic vectors using the canonical correlations from step (13) as weights. These are the estimated initial weights.

(19) Subroutine DORMF - The resulting weights from (18) are normalized.

(20) Subroutine FILUP - This subroutine packs the normalized estimated initial weights into a single-dimensioned array.

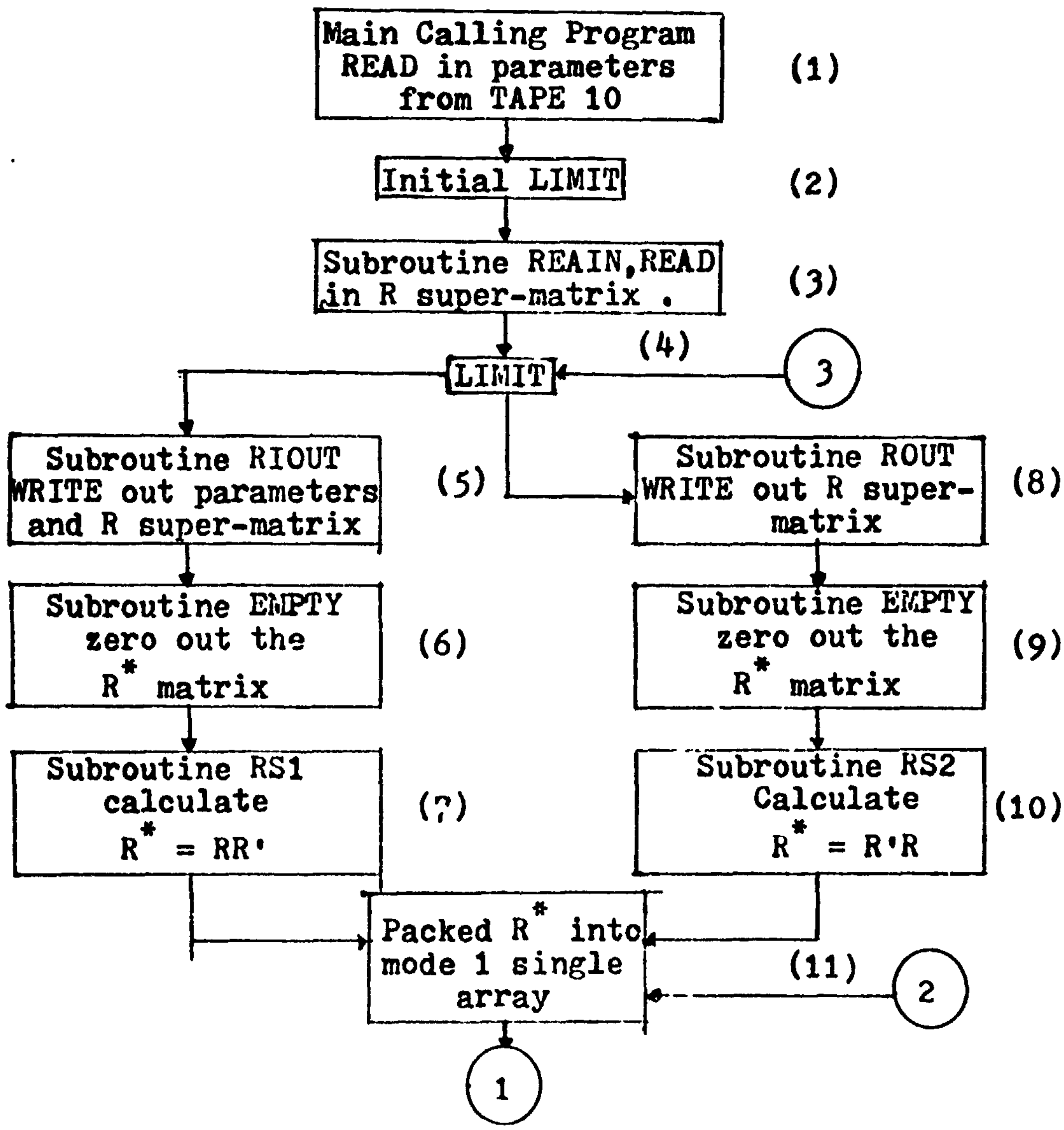
(21) WRITE the normalized estimated initial weignts which are calculated from (19), for each set. The packed initial weights from FILUP, will replace the

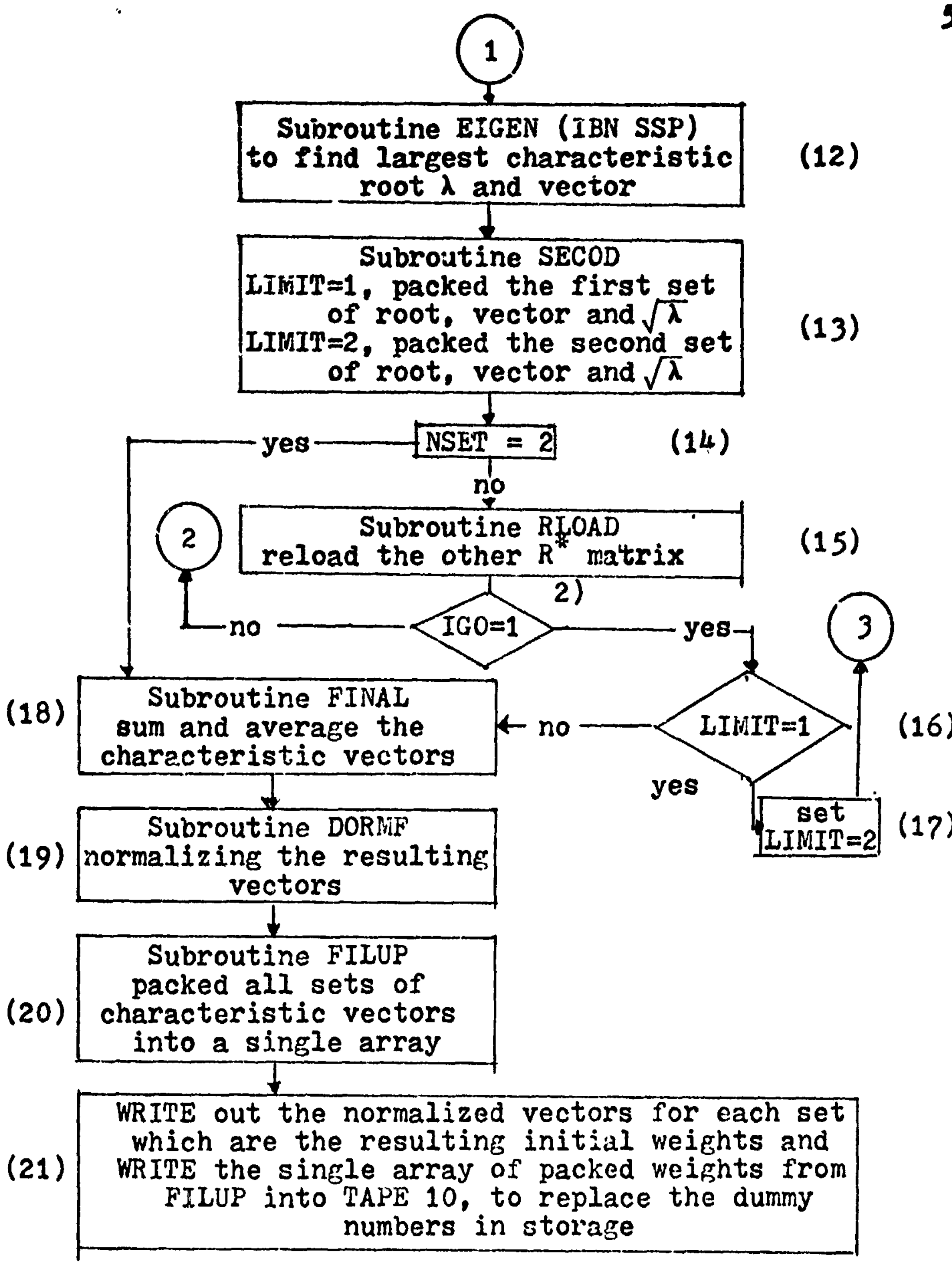
dummy numbers in storage- the 3rd record in TAPE 10.

Some intermediate results are stated in section 6.6; however, since these were explanatory programs, no detailed illustrations have been provided. CHAPTER VI contains several illustrations using the best initialization, i.e. the canonical-multiple weights.

Computer Program ICW

Flow-Chart





2) IGO = 2 refers to the left pass of the previous page.
IGO = 1 refers to the right pass.
It is reset by the Subroutine RLOAD.

4.3 Minimum Determinant Solution

4.3.1 Description of Algorithm

We want to find weights $\underline{a}_i$ $(i = 1, 2, \ldots, p)$ such that a variable (canonical variable) can be formed for each response set.

$$u_i = \underline{a}_i'\underline{y}_i \quad (i = 1, 2, \ldots, p) \tag{4.3.1.1}$$

with $\operatorname{corr}(\underline{a}_i'\underline{y}_i, \underline{y}_j'\underline{a}_j) = \operatorname{corr}(u_i, u_j) = \rho_{ij}$

$$\underline{a}_i' R_{ij} \underline{a}_j \tag{4.3.1.2}$$

under the contraint $\underline{a}_i' R_{ii} \underline{a}_i = \underline{a}_i'\underline{a}_i = 1$ for all i's and P is the matrix with typical element ρ_{ij}.

Formally,

$$\begin{bmatrix} 1 & \underline{a}_1' R_{12} \underline{a}_2 & \cdots & \underline{a}_1' R_{1p} \underline{a}_p \\ \underline{a}_2' R_{12}' \underline{a}_1 & 1 & \cdots & \underline{a}_2' R_{2p} \underline{a}_p \\ \cdot & \cdot & \cdots & \cdot \\ \underline{a}_p' R_{1p}' \underline{a}_1 & \underline{a}_p' R_{2p} \underline{a}_2 & \cdots & 1 \end{bmatrix} \tag{4.3.1.3}$$

a $(p \times p)$ matrix. We need to find the vector $\underline{a}_i$ which minimize $|P|$ (we use $\ln|P|$, for simplicity). Now

$$\frac{\partial \ln|P|}{\partial a_{1i}} = \operatorname{tr} \frac{\partial \ln|P|}{\partial P} \frac{\partial P}{\partial a_{1i}} \text{ [3]}$$

$$= \operatorname{tr} P^{-1} \begin{bmatrix} (2\underline{a}_1)_i & (R_{12}\underline{a}_2)_i & \cdots & (R_{1p}\underline{a}_p)_i \\ (\underline{a}_2' R_{12}')_i & 0 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ (\underline{a}_p' R_{1p}')_i & 0 & \cdots & 0 \end{bmatrix} \tag{4.3.1.4}$$

---

3) Where $(R_{11}\underline{a}_1)_i$ denotes the i^{th} element of $R_{11}\underline{a}_1$.

Denote the inverse of P by

$$P^{-1} = \begin{bmatrix} \rho^{11} & \rho^{12} & \cdots & \rho^{1p} \\ \rho^{12} & \rho^{22} & \cdots & \rho^{2p} \\ \cdot & \cdot & \cdots & \cdot \\ \rho^{1p} & \rho^{2p} & \cdots & \rho^{pp} \end{bmatrix} \qquad (4.3.1.5)$$

Simplifying $\frac{\partial \ln |P|}{\partial a_{1i}}$, we have

$$\frac{\partial \ln |P|}{\partial a_{1i}} = \mathrm{tr} \begin{bmatrix} \rho^{11} & \rho^{12} & \cdots & \rho^{1p} \\ \cdot & \cdot & \cdots & \cdot \\ \rho^{1p} & \rho^{2p} & \cdots & \rho^{pp} \end{bmatrix} \begin{bmatrix} (2\,\underline{a}_1)_i & \cdots & (R_{1p}\underline{a}_p)_i \\ \cdot & \cdots & \cdot \\ (\underline{a}_p' R_{1p}')_i & \cdots & 0 \end{bmatrix} \qquad (4.3.1.6)$$

Hence, in general

$$\frac{\partial \ln |P|}{\partial \underline{a}_1} = 2\,\rho^{11}\underline{a}_1 + 2\,\rho^{12}R_{12}\underline{a}_2 + \ldots + 2\,\rho^{1p}R_{1p}\underline{a}_p$$

$$\frac{\partial \ln |P|}{\partial \underline{a}_2} = 2\,\rho^{12}R_{12}'\underline{a}_1 + 2\,\rho^{22}\underline{a}_2 + \ldots + 2\,\rho^{2p}R_{2p}\underline{a}_p$$

$$\cdots \qquad \cdots \qquad \cdots \qquad \cdots \qquad \cdots$$

$$\frac{\partial \ln |P|}{\partial \underline{a}_p} = 2\,\rho^{1p}R_{1p}'\underline{a}_1 + 2\,\rho^{2p}R_{2p}'\underline{a}_2 + \ldots + 2\,\rho^{pp}\underline{a}_p \quad ; \qquad (4.3.1.7)$$

the constraint is $\underline{a}_i'\underline{a}_i = 1$ for all i's. By the method of Lagrangian multipliers, taking the partial derivatives with respect to $\underline{a}_i$ and setting them to zero; for i = 1 we find

$$\frac{\partial[\ln|P| - \sum_{i=1}^{p}\lambda_i(\underline{a}_i' R_{ii}\underline{a}_i - 1)]}{\partial \underline{a}_i}$$

$$= 2\rho^{11}\underline{a}_1 + 2\rho^{12}R_{12}\underline{a}_2 + \dots + 2\rho^{1p}R_{1p}\underline{a}_p$$

$$- 2\lambda_1\underline{a}_1 \tag{4.3.1.8}$$

To determine the Lagrangian multiplier we premultiply by $\underline{a}_i'$, obtaining

$$\rho^{11}\rho_{11} + \rho^{12}\rho_{12} + \dots + \rho^{1p}\rho_{1p} - \lambda_1\rho_{11} = 0 \tag{4.3.1.9}$$

but $\sum_{i=1}^{p}\rho^{1i}\rho_{1i} = 1$ and $\rho_{11} = 1$, therefore

$$\lambda_1 = 1$$

The derivatives with respect to the other $\underline{a}_i$'s are obtained in the same manner.

A system of equations is thus set up in the following matrix form:

$$\begin{bmatrix} (\rho^{11}-1)\,I & \rho^{12}R_{12} & \dots & \rho^{1p}R_{1p} \\ \rho^{12}R_{12}' & (\rho^{22}-1)\,I & \dots & \rho^{2p}R_{2p} \\ \cdot & \cdot & \dots & \cdot \\ \rho^{1p}R_{1p}' & \rho^{2p}R_{2p}' & \dots & (\rho^{pp}-1)\,I \end{bmatrix} \begin{bmatrix} \underline{a}_1 \\ \underline{a}_2 \\ \cdot \\ \underline{a}_p \end{bmatrix} = \begin{bmatrix} \underline{0} \\ \underline{0} \\ \cdot \\ \underline{0} \end{bmatrix} \tag{4.3.1.10}$$

The problem is a rather complicated generalization of a characteristic root-vector problem. We must find element ρ^{ij} $(i = 1, 2, \dots, p;\ j = 1, 2, \dots, p)$ such that the determinant of the super-matrix (4.3.1.10) is zero and such that ρ^{ij} are elements of an inverse of a correlation matrix, i.e. they form a positive-definite matrix whose principal cofactors are all equal to each other, and to the determinant

of P^{-1}.

By solving this system of equations (for non-trivial solutions) we have the set of canonical weights which maximize our measure of overall correlation between all categorical variables.

In the numerical solution, we employ the descent method of R. Fletcher and M.J.D. Powell [1963] to minimize the function $\ln|P|$ and obtain the canonical weights.

Since we cannot use the Lagrange method directly , we obtain the derivatives for the uncontrained problem:

The unconstrained correlation matrix P has typical element

$$\rho_{ij} = \frac{\underline{a}_i' R_{ij} \underline{a}_j}{\ell_i \ell_j}, \tag{4.3.1.11}$$

where $\ell_i = \sqrt{\underline{a}_i'\underline{a}_i}$ and $\ell_j = \sqrt{\underline{a}_j'\underline{a}_j}$

Hence

$$P = \begin{bmatrix} \frac{\underline{a}_1'\underline{a}_1}{\ell_1^2} & \frac{\underline{a}_1'R_{12}\underline{a}_2}{\ell_1\ell_2} & \cdots & \frac{\underline{a}_1'R_{1p}\underline{a}_p}{\ell_1\ell_p} \\ \frac{\underline{a}_2'R_{12}'\underline{a}_1}{\ell_1\ell_2} & \frac{\underline{a}_2'\underline{a}_2}{\ell_2^2} & \cdots & \frac{\underline{a}_2'R_{2p}\underline{a}_p}{\ell_2\ell_p} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\underline{a}_p'R_{1p}'\underline{a}_1}{\ell_1\ell_p} & \frac{\underline{a}_p'R_{2p}'\underline{a}_2}{\ell_2\ell_p} & \cdots\cdot & \frac{\underline{a}_p'\underline{a}_p}{\ell_p^2} \end{bmatrix} \tag{4.3.1.12}$$

We take the derivatives of $\ln|P|$ with respect to $\underline{a}_m$ where m is a fixed subscript ($m = 1, 2, \ldots, p$):

$$\frac{\partial \ln|P|}{\partial \underline{a}_m} = \sum_{ij}\sum \frac{\partial \ln|P|}{\partial \rho_{ij}} \frac{\partial \rho_{ij}}{\partial \underline{a}_m}$$

since

$$\frac{\partial \ln|P|}{\partial \rho_{ij}} = \rho^{ij} \quad \text{and}$$

$$\frac{\partial \rho_{ij}}{\partial \underline{a}_m} = \frac{\partial \dfrac{\underline{a}_i' R_{ij} \underline{a}_j}{\ell_i \, \ell_j}}{\partial \underline{a}_m}$$

$$= \left[\delta_i^m \frac{1}{\ell_i \ell_j} R_{ij}\underline{a}_j + \delta_j^m \frac{1}{\ell_i \ell_j} R_{ij}'\underline{a}_i\right]$$

$$- \left[\delta_i^m \frac{\rho_{ij}}{\ell_i \ell_j} \frac{\ell_j}{\ell_i} a_i + \delta_j^m \frac{\rho_{ij}}{\ell_i \ell_j} \frac{\ell_i}{\ell_j} a_j\right]$$

where δ_i^m is the Kronecker delta.

Hence

$$\frac{\partial \ln|P|}{\partial \underline{a}_m} = \sum_{ij}\sum \left\{\left[\delta_i^m \frac{\rho_{ij}}{\ell_i \ell_j} R_{ij}\underline{a}_j + \delta_j^m \frac{\rho_{ij}}{\ell_i \ell_j} R_{ij}'\underline{a}_i\right]\right.$$

$$\left. - \left[\delta_i^m \frac{\rho_{ij}\,\rho^{ij}}{\ell_i \, \ell_j} \frac{\ell_j}{\ell_i} \underline{a}_i + \delta_j^m \frac{\rho_{ij}\,\rho^{ij}}{\ell_i \, \ell_j} \frac{\ell_i}{\ell_j} \underline{a}_j\right]\right\}$$

$$= \left[\sum_j \frac{\rho^{mj}}{\ell_m \ell_j} R_{mj}\underline{a}_j + \sum_i \frac{\rho^{im}}{\ell_i \ell_m} R_{im}'\underline{a}_i\right]$$

$$- \left[\sum_j \frac{\rho_{mj}\,\rho^{mj}}{\ell_m^2} \underline{a}_m + \sum_i \frac{\rho_{im}\,\rho^{im}}{\ell_m^2} \underline{a}_m\right]$$

$$= \frac{1}{\ell_m}\left[\sum_j \frac{\rho^{mj}}{\ell_j} R_{mj}\underline{a}_j + \sum_i \frac{\rho^{im}}{\ell_i} R'_{im}\underline{a}_i\right]$$

$$- \frac{1}{\ell_m^2}\underline{a}_m\left[\sum_j \rho_{mj}\rho^{mj} + \sum_i \rho_{im}\rho^{im}\right]$$

$$= \frac{2}{\ell_m}\sum_j \frac{\rho^{mj}}{\ell_j} R_{mj}\underline{a}_j - \frac{2}{\ell_m^2}\underline{a}_m \qquad (4.3.1.14)$$

since $\sum_j \rho_{mj}\rho^{mj} = \sum_i \rho_{im}\rho^{im} = 1$,

$$\frac{\partial \ln |P|}{\partial \underline{a}_m} = \frac{2}{\ell_m}\sum_j \frac{\rho^{mj}}{\ell_j} R_{mj}\underline{a}_j - \frac{2}{\ell_m^2}\underline{a}_m \qquad (4.3.1.15)$$

We must solve these equations in order to minimize $\ln |P|$ and to obtain the canonical weights.

To obtain the gradient of $\ln |P|$ we evaluate the $(\ell_1 + \ell_2 + \ldots + \ell_k - k)$ expressions

$$\frac{\partial \ln |P|}{\partial \underline{a}_m} = \frac{2}{m}\left[\frac{\rho^{m1}}{\ell_1} R_{m1}\underline{a}_1 + \ldots + \frac{\rho^{m,m-1}}{\ell_{m-1}} R_{m,m-1}\underline{a}_{m-1}\right.$$

$$\left. + \frac{(\rho^{mm}-1)}{\ell_m}\underline{a}_m + \ldots + \frac{\rho^{mk}}{\ell_k} R_{mk}\underline{a}_k\right] \qquad (4.3.1.16)$$

since $R_{mm} = I$ and $\underline{a}_m$ has $(\ell_m - 1)$ components.

Subroutine FUNCT places $\ln |P|$ into F, and the gradient vector, calculated by the above formula, and into G.

4.3.2 Description of the Computer Program for FPM

This is the computer program for obtaining Minimum Determinant Solution, in brief: FPM.

The listing of this program is in Appendix C; the data are stored in TAPE 10. This program proceeds as follow:

(1) The main calling program has the following operations:

(a) Call Subroutine IO, for data INPUT from TAPE 10, the temporary storage.

(b) Call Subroutine FMFP (IBM SSP) to find the minimum of $\ln |P|$.

(c) After the Subroutines have been completed, the main calling program will output:

(i) The canonical weights.

(ii) The determinant of the calculated correlation matrix, i.e. $|P|$.

(iii) The gradients of $\ln |P|$.

(iv) Canonical weights in normalized form.

(Error codes, if any, and the number of iterations needed are also stated as output.)

(2) Subroutine IO: This Subroutine will READ from the TAPE 10, all needed information, and output:

N - Number of variables involved.

LIMIT - Maximum number of iterations.

EPS - Permissible error during iterations.

EST - Estimated minimum of the given function.

R - The super-matrix which has been produced from C-E-R program, in the form of a row-wise list.

The Subroutine CONEC is called to construct an array from the super-matrix R, needed for the Subroutine FMFP.

(3) Subroutine FMFP: This Subroutine is from IBM SSP (Scientific Subroutine Package) and was developed from the Fletcher and Powell process (System/360-CM-03X Version III - H20-0205-3, 223).[3)]

FUNCT - User written subroutine for minimizing given function and calculate gradients.

N - Number of variables.

X - Vector containing initial weights, and it will contain the final result.

F - A single variable containing the minimum function value.

G - Vector containing the gradients.

EST - Estimated minimum function value.

EPS - Expected absolute error.

LIMIT - Maximum number of iterations.

IER - Error codes:

IER = 0, convergence was obtained.

IER = 1, no convergence in LIMIT iterations.

---

3) But an error in the IBM SSP program had to be corrected first.

IER = -1, error in gradient calculation.

IER = 2, no minimum exists.

H - Working storage.

The Subroutine FUNCT is the user written subroutine for $\ln |P|$ to be minimized. Our FUNCT subroutine will call Subroutine MINV (another IBM SSP) for finding a matrix inverse, needed for the intermediate calculations.

This user written subroutine FUNCT is called by the subroutine FMFP (IBM SSP) whenever a new set of canonical weights is determined. Within the FUNCT the correlation matrix P is calculated for each new set of canonical weights then the subroutine MINV (IBM SSP) is called to find P^{-1}, the two-dimensional array of P is packed into a single-dimensioned array as INPUT argument for MINV (since P is in storage mode 1, pnly the upper triangular part of P is packed), as the P^{-1} is found through MINV, and back into the FUNCT the gradient is calculated for determining the convergency of the given function.

In the correlation matrix, the (i, j) element is

$$\rho_{ij} = \frac{\underline{a}_i' R_{ij} \underline{a}_j}{\sqrt{\underline{a}_i' \underline{a}_i} \sqrt{\underline{a}_j' \underline{a}_j}} \qquad (4.3.2.1)$$

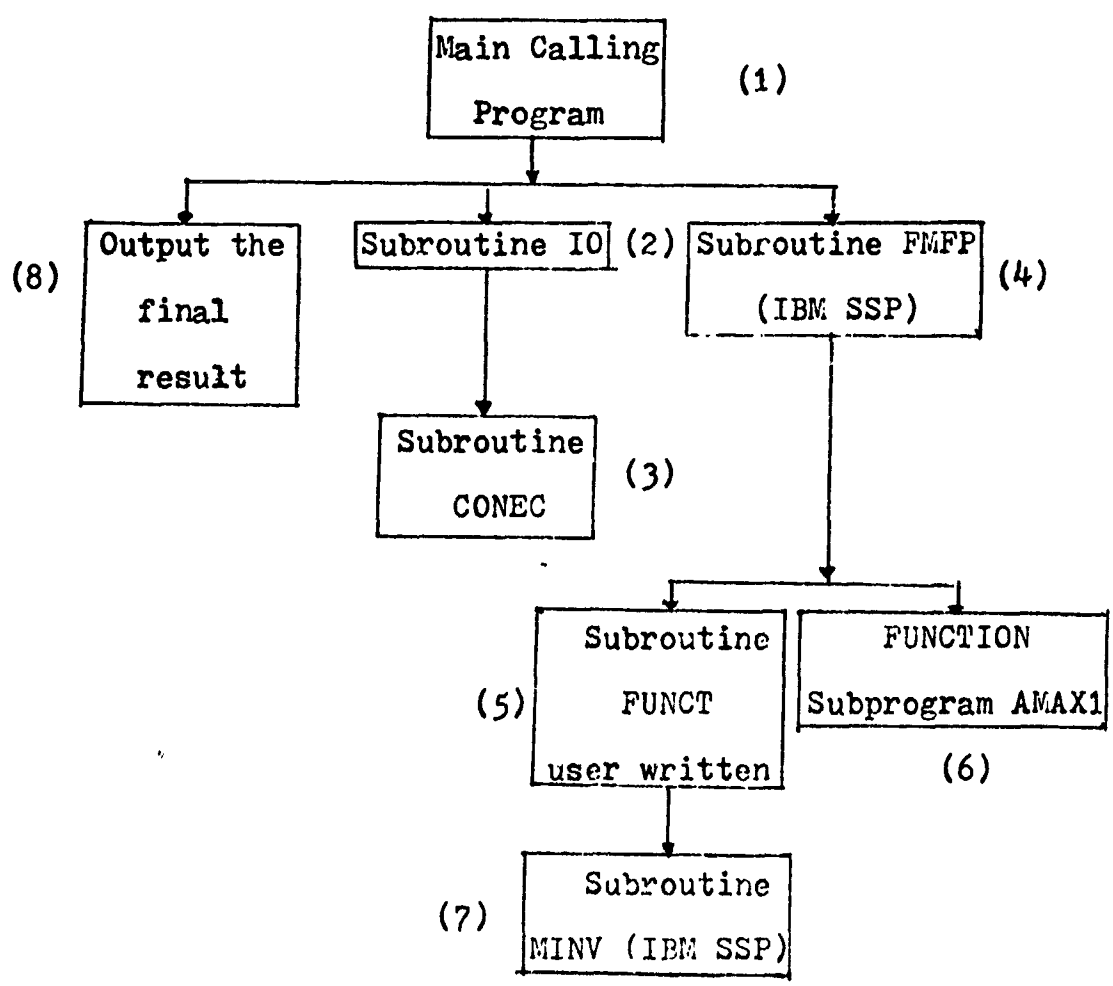
and for the gradient, it is the partial derivatives of $\ln |P|$ with respect to $\underline{a}_m$ (the m^{th} categorical weights), hence

$$\frac{\partial \ln |P|}{\partial \underline{a}_m} = \frac{2}{\ell_m}\left[\sum_j \frac{\rho^{mj} R_{mj} \underline{a}_j}{\ell_j} - \frac{\underline{a}_m}{\ell_m}\right] \qquad (4.3.2.2)$$

where $\ell_m = \sqrt{\underline{a}_m' \underline{a}_m}$ and $\ell_j = \sqrt{\underline{a}_j' \underline{a}_j}$, and at the same time, the value of $\ln |P|$ is calculated.

Computer Program FPM

Flow-Chart



4.4 Reduction to Original Scales

4.4.1 Description of Algorithm

The $\underline{a}_i$ weights found by the Minimum Determinant Solution must now be translated into the desired categorical scales based upon the E matrix (section 4.1.1).

For the choice of standardized scales, we introduce the scalar

$$k_i = \frac{\underline{n}'_{stuvq}\underline{w}_i}{n} \qquad (4.4.1.1)$$

where $\underline{n}'_{stuvq}$ is a vector of marginal sums, viz.,

$$\underline{n}'_{s....} = [n_{1....}, n_{2....}, \ldots, n_{r_1....}]$$

$$\underline{n}'_{.t...} = [n_{.1...}, n_{.2...}, \ldots, n_{.r_2...}]$$

$$\underline{n}'_{..u..} = [n_{..1..}, n_{..2..}, \ldots, n_{..r_3..}]$$

$$\underline{n}'_{...v.} = [n_{...1.}, n_{...2.}, \ldots, n_{...r_4.}]$$

$$\underline{n}'_{....q} = [n_{....1}, n_{....2}, \ldots, n_{....r_5}] \qquad (4.4.1.2)$$

$$\underline{w}'_i = \underline{a}'_i T_i^{(-1)} \qquad (4.4.1.3)$$

$T_i^{(-1)}$ is a conditional inverse found in the program C-E-R (section 4.1.1).

$$n = \text{grand total} = \underline{n}'_{s....}\underline{1} = \underline{n}'_{.t...}\underline{1} = \underline{n}'_{..u..}\underline{1}$$
$$= \underline{n}'_{...v.}\underline{1} = \underline{n}'_{....q}\underline{1} \qquad (4.4.1.4)$$

$$\underline{1}' = [1, 1, \ldots, 1] \qquad (4.4.1.5)$$

$n_{.....}$ are marginal totals which are obtained in the process of construction of the E matrix.

We now form a new vector $\underline{w}_i^*$ by subtracting k_i from each

element of $\underline{w}_i$, i.e.

$$\underline{w}_i^{*'} = \underline{w}_i - k_i\underline{1} \; ; \qquad (4.4.1.6)$$

and thus

$$\begin{aligned} \underline{n}'_{s\ldots\ldots}\underline{w}_i^{*} &= n_{1\ldots\ldots}w_{11}^{*} + n_{2\ldots\ldots}w_{12}^{*} + \ldots \\ &\quad + n_{r_1\ldots\ldots}w_{1r_1}^{*} \\ &= n_{1\ldots\ldots}(w_{11} - k_1) + n_{2\ldots\ldots}(w_{12} - k_1) \\ &\quad + \ldots + n_{r_1\ldots\ldots}(w_{1r_1} - k_1) \\ &= \underline{n}'_{s\ldots\ldots}\underline{w}_1 - nk_1 = nk_1 - nk_1 = 0 \qquad (4.4.1.7) \end{aligned}$$

and similarily

$$\underline{n}'_{.t\ldots}\underline{w}_2^{*} = \underline{n}'_{..u..}\underline{w}_3^{*} = \underline{n}'_{...v.}\underline{w}_4^{*} = \underline{n}'_{....q}\underline{w}_5^{*} = 0 \qquad (4.4.1.8)$$

This is consistent since each of the rows and columns of every E_{ij} matrix sums up to zero.

Now let

$$p_i = \frac{\underline{n}'_{stuvq}\underline{w}_i^{*(2)}}{n} \qquad (4.4.1.9)$$

where $\underline{w}_i^{*(2)} = [w_{i1}^{*2}, w_{i2}^{*2}, \ldots w_{ir_i}^{*2}]$ and

$\underline{n}'_{stuvq}$ is defined as in (4.4.1.2).

Let

$$\underline{w}_i^{**} = \frac{\underline{w}_i^{*}}{p_i} \quad \text{then} \quad \underline{w}_i^{**(2)} = \frac{\underline{w}_i^{*(2)}}{p_i} \qquad (4.4.1.10)$$

hence $\underline{w}_i^{**}$ for all i's are the categorical weight vectors for the entire E matrix.

Since

$$\underline{n}'_{stuvq}\underline{w}_i^{**(2)} = \frac{\underline{n}'_{stuvq}\underline{w}_i^{*(2)}}{p_i} = \frac{np_i}{p_i} = n \qquad (4.4.1.11)$$

therefore

$$\frac{\underline{n}'_{stuvq}\underline{w}_i^{**}}{n} = \frac{n}{n} = 1 \qquad (4.4.1.12)$$

We conclude that, if each of the original categorical responses is translated into a new set of scaled weights obtained in this manner then, for the sample at hand, the mean of the scaled weights will be zero and the standard deviation will be one.

Now we let, for j^{th} subject

$$u_{ij} = \underline{w}_i' \, \underline{y}_{ij}^{**} \quad \text{and} \quad u_{kj} = \underline{w}_k' \, \underline{y}_{kj}^{**} \qquad (4.4.1.13)$$

it remains to be shown that the $\underline{w}_i$ can be replaced by $\underline{w}_i^{**}$ without change of the correlation between resulting canonical variables.

Let there be s factors (response variables), let factor 1 have r_1 levels (i.e. variable 1 has r_1 distinct categorical states), factor 2 have r_2 levels, ... and factor s have r_s levels. In analogy with the Fisher-Lancaster approach, we introduce $r_1 + r_2 + \ldots + r_s$ pseudo-variables. Each response vector consisting of s categorical responses (one to each factor), is translated into a vector of 0's and 1's; if the response was $[c_1, c_2, \ldots, c_s]$ the resulting vector will be a rolled out row vector, consisting of s parts; the first part will have a 1 in position c_1 and zeros elsewhere; the s^{th} part will have a 1 in position c_s, and zeros elsewhere. For example: if there were four factors with [3, 4, 2, 5] levels respectively, and if an experimental unit had a response of [2, 1, 2, 3], the corresponding y-vector would be

010 1000 01 00100 .

The E matrix consists of the matrices of corrected sums of squares and products of these observations. Let $\underline{y}'_{ij}$ denote the i^{th} part ($i = 1, 2, \ldots, s$) of this vector for subject No. j. Let $\bar{\underline{y}}'_i$ denote the average over the entire sample, obviously

$$\bar{\underline{y}}'_1 = \frac{1}{n} [\, n_{1\ldots}, n_{2\ldots}, \ldots, n_{r_1\ldots}]$$

$$\bar{\underline{y}}'_2 = \frac{1}{n} [\, n_{.1\ldots}, n_{.2\ldots}, \ldots, n_{.r_2\ldots}]$$

$$\ldots \qquad (4.4.1.14)$$

etc.

Let $\underline{y}^{*\prime}_{ij} = \underline{y}'_{ij} - \bar{\underline{y}}'_i$, then $E_{ii} = \sum_{j=1}^{s} \underline{y}^{*}_{ij}\, \underline{y}^{*\prime}_{ij}$ (4.4.1.15)

the matrix of sums of squares and products of the elements of $\underline{y}^{*\prime}_{ij}$, and

$$E_{ik} = \sum_{j=1}^{s} \underline{y}^{*}_{ij}\, \underline{y}^{*\prime}_{kj} \qquad (4.4.1.16)$$

Now we let $u_{ij} = \underline{w}'_i\, \underline{y}^{*}_{ij}$ and $u_{kj} = \underline{w}'_k\, \underline{y}^{*}_{kj}$ represent two scaled responses for subject j. Over the entire sample, we can thus calculate a canonical correlation between scaled responses i and k, as

$$\begin{aligned}
&\underline{w}^{*\prime}_i \sum_{j=1}^{s} \underline{y}^{*}_{ij}\, \underline{y}^{*\prime}_{kj}\, \underline{w}^{*}_k + k_i \underline{1}' \sum_{j=1}^{s} \underline{y}^{*}_{ij}\, \underline{y}^{*\prime}_{kj}\, \underline{w}^{*}_k \\
&\qquad + \underline{w}^{*\prime} \sum_{j=1} \underline{y}^{*}_{ij}\, \underline{y}^{*\prime}_{kl}\, k_k \underline{1} + k_i k_k \underline{1}' \sum_{j=1}^{s} \underline{y}^{*}_{ij}\, \underline{y}^{*\prime}_{kj}\, \underline{1} \\
&\qquad = \underline{w}^{*\prime}_i E_{ik}\, \underline{w}^{*}_k + k_i \underline{1}'\, E_{ik}\, \underline{w}^{*}_k + \underline{w}^{*\prime}_i E_{ik}\, k_k \underline{1} + k_i k_k \underline{1}' E_{ik}\, \underline{1} \\
&\qquad = \underline{w}^{*\prime}_i E_{ik}\, \underline{w}^{*}_k \qquad (4.4.1.17)
\end{aligned}$$

where the k_i's are defined in (4.4.1.1).

Since all row and column sums of E_{ik} are zero, the canonical correlation between U_i and U_k remains unchanged if each $\underline{w}_i$ is replaced by $\underline{w}_i^*$. And since $\underline{w}_i^{**}$ differ from $\underline{w}_i^*$ only by a scalar multiplier, the canonical correlation between U_i and U_k is obviously unchanged if $\underline{w}_i^*$ is replaced by $\underline{w}_i^{**}$.

Recall that $R = T^{(-1)}E\,T^{(-1)'}$, where

$$T^{(-1)} = \begin{bmatrix} T_1^{(-1)} & 0 & \cdots & 0 \\ 0 & T_2^{(-1)} & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdots & T_s^{(-1)} \end{bmatrix} \qquad (4.4.1.18)$$

hence $R_{ik} = T_i^{(-1)}E_{ik}\,T_k^{(-1)'}$ and

$$\sum_{k=1}^{s} (\rho^{ik} - \delta_{ik})R_{ik}\underline{a}_k = \underline{0}$$

where δ_{ik} is the Kronecker delta (from 4.3.1.10).

Then $\sum_{k=1}^{s} (\rho^{ik} - \delta_{ik})\,T_{ik}^{(-1)}E_{ik}\,T^{(-1)'}\underline{a}_k = \underline{0}$. (4.4.1.19)

Let $T_k^{(-1)}\,\underline{a}_k = \underline{w}_k$, hence

$$T_i^{(-1)}\sum_{k=1}^{s} (\rho^{ik} - \delta_{ik})\,E_{ik}\,\underline{w}_k = \underline{0} \qquad (4.4.1.20)$$

thus the $\underline{w}_k$ are the weights applicable to the unreduced matrices E_{ik} ($i = 1, 2, \ldots, s$; $k = 1, 2, \ldots, s$) and therefore to the original variables.

4.4.2 Description of the Computer Program for RTE

This is the computer program for obtaining the original categorical scales for E matrix, in brief: RTE. The listing

of this computer program is in Appendix E. And this program proceeds as follow:

(1) Data stored in TAPE 10, are read. In this program, we need the following information:

Record 1: NWTS - Number of weights.

Record 2: NSETS - Number of sets.

(NRST(I),I=1,NSETS) - Number of rows in each set.

Record 3: (X(I),I=1,NWTS) - Canonical weights from the Minimum-Determinant Solution.

Record 4: ((NT(JA,I),I=1,IA),JA=1,NSETS) - Marginal totals for each set.

Record 5: NTAL - The grand total.

Record 6: T(I,J,K) - T conditional inverses.

(2) Packed $\underline{a}_i$ for all i's, the canonical weights into a two-dimensional array for calculating $\underline{w}_i$.

(3) Calculate:

(a) $$\underline{w}_i' = \underline{a}_i' \; T_i^{(-1)} \qquad (4.4.2.1)$$

(b) $$k_i = \frac{\underline{n}_i' \; \underline{w}_i}{n} \qquad (4.4.2.2)$$

(c) $$\underline{w}_i^* = \underline{w}_i - k_i\underline{1} \qquad (4.4.2.3)$$

(d) $$p_i = \frac{n_1 w_{i1}^{*2} + n_2 w_{i2}^{*2} + \ldots + n_s w_{is}^{*2}}{n} \qquad (4.4.2.4)$$

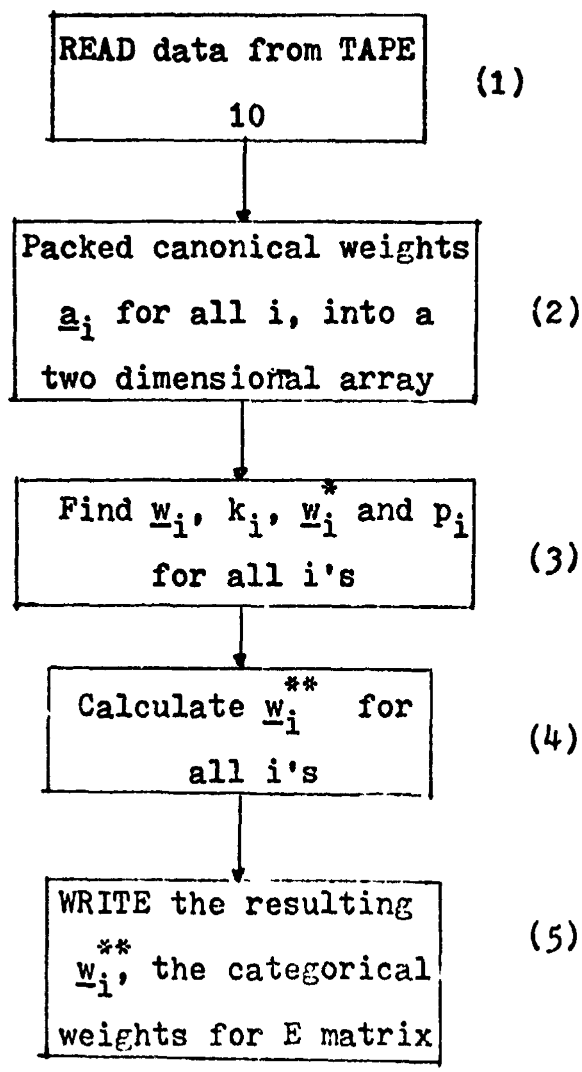
(4) Calculate:

$$\underline{w}_i^{**} = \frac{\underline{w}_i^*}{p_i} \qquad (4.4.2.5)$$

(5) WRITE the resulting categorical weights for E matrix, i.e. $\underline{w}_i^{**}$ for all i's.

Computer Program RTE

Flow-Chart



CHAPTER V

RELATION TO INVERSE OF CANONICAL-PARTIAL AND CANONICAL-MULTIPLE CORRELATION MATRICES

It is well known that the diagonal elements of the inverse of a correlation matrix are $\dfrac{1}{(1 - \rho^2_{i.rest})}$, where $\rho_{i.rest}$ is the multiple correlation between the i^{th} variable and the others in the set. Also, it is well known that normalization of the inverse matrix into a correlation matrix and change of all signs in the off-diagonal elements produces the matrix of partial correlations of each pair of variables, given the rest. Hence in the inverse matrix, the off-diagonal elements are

$$\frac{-\rho_{ij.rest}}{\sqrt{(1 - \rho^2_{i.rest})\,(1 - \rho^2_{j.rest})}} \qquad (5.1)$$

Using this structure of an inverse we may, analogously, calculate canonical-partial and canonical-multiple correlations and produce a matrix which could be regarded as the inverse of the matrix P. This procedure is outlined below.

We constructed an E matrix for some categorical variables with different responses and transformed the E matrix into a rank-reduced super-matrix R with the diagonal matrices equal to I (the identity matrix). Hence

$$R = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1p} \\ R'_{12} & R_{22} & \cdots & R_{2p} \\ \cdot & \cdot & \cdots & \cdot \\ R'_{1p} & R'_{2p} & \cdots & R_{pp} \end{bmatrix} \tag{5.2}$$

where $R_{11} = R_{22} = \ldots = R_{pp} = I$.

We first partitioned the super-matrix R

$$\left[\begin{array}{c|c|cccc} I & R_{12} & R_{13} & \cdots & R_{1p} \\ \hline R'_{12} & I & R_{23} & \cdots & R_{2p} \\ \hline R'_{13} & R'_{23} & I & \cdots & R_{3p} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ R'_{1p} & R'_{2p} & R'_{3p} & \cdots & I \end{array}\right] \tag{5.3}$$

and then obtained a new matrix

$$R^* = \begin{bmatrix} R^*_{11} & R^*_{12} \\ R^{*\prime}_{12} & R^*_{22} \end{bmatrix} \tag{5.4}$$

where $$R^*_{11} = I - \begin{bmatrix} R_{13} & \cdots & R_{1p} \end{bmatrix} \begin{bmatrix} I & \cdots & R_{3p} \\ \cdot & \cdots & \cdot \\ R'_{3p} & \cdots & I \end{bmatrix}^{-1} \begin{bmatrix} R'_{13} \\ \cdot \\ R'_{1p} \end{bmatrix}$$

$$R^*_{22} = I - \begin{bmatrix} R_{23} & \cdots & R_{2p} \end{bmatrix} \begin{bmatrix} I & \cdots & R_{3p} \\ \cdot & \cdots & \cdot \\ R'_{3p} & \cdots & I \end{bmatrix}^{-1} \begin{bmatrix} R'_{23} \\ \cdot \\ R'_{2p} \end{bmatrix}$$

and $$R^*_{12} = R_{12} - \begin{bmatrix} R_{13} & \cdots & R_{1p} \end{bmatrix} \begin{bmatrix} I & \cdots & R_{3p} \\ \cdot & \cdots & \cdot \\ R'_{3p} & \cdots & I \end{bmatrix}^{-1} \begin{bmatrix} R'_{23} \\ \cdot \\ R'_{2p} \end{bmatrix}$$

The square of the canonical-partial correlation of set 1 and set 2 given the others is the largest characteristic root of $R^{*-1}_{11} R^*_{12} R^{*-1}_{22} R^{*'}_{12}$, i.e.

$$\rho^2_{12.34\ldots p} = Ch_{max}(R^{*-1}_{11} R^*_{12} R^{*-1}_{22} R^{*'}_{12}) \tag{5.5}$$

But for convenience in calculating the characteristic root, we first find a T^* such that $R^*_{11} = T^* T^{*'}$ and then invert T^*, obtaining T^{*-1}; then $R^{*-1}_{11} = (T^{*-1})'(T^{*-1})$. Since $Ch(ABC) = Ch(CAB)$ then

$$\begin{aligned} \rho^2_{12.34\ldots p} &= Ch_{max}(R^{*-1}_{11} R^*_{12} R^{*-1}_{22} R^{*'}_{12}) \\ &= Ch_{max}(T^{*-1'} T^{*-1} R^*_{12} R^{*-1}_{22} R^{*'}_{12}) \\ &= Ch_{max}(T^{*-1} R^*_{12} R^{*-1}_{22} R^{*'}_{12} T^{*-1'}) \end{aligned} \tag{5.6}$$

in which the matrix inside Ch(...) is symmetric and real. By the same process we can obtain all of the sample canonical-partial correlation between any two sets of categorical variables given the others, by taking the square root of the characteristic roots

$$\rho_{13.24\ldots p}, \; \cdots, \; \rho_{1p.234..(p-1)}, \; \cdots, \; \rho_{(p-1)p.123\ldots(p-2)}. \tag{5.7}$$

We can place these canonical-partial correlations into a (p x p) matrix (off-diagonal terms).

For the canonical-multiple correlations, We partitioned the super-matrix R

$$\left[\begin{array}{c|cccc} I & R_{12} & R_{13} & \cdots & R_{1p} \\ \hline R'_{12} & I & R_{23} & \cdots & R_{2p} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ R'_{1p} & R'_{2p} & R'_{3p} & \cdots & I \end{array}\right] = \begin{bmatrix} I & R^{*}_{12} \\ R^{*\prime}_{12} & R^{*}_{22} \end{bmatrix} \qquad (5.8)$$

where $R^{*}_{12} = \quad R_{12} \quad R_{13} \quad \cdots \quad R_{1p}$,

$$R^{*}_{22} = \begin{bmatrix} I & R_{23} & \cdots & R_{2p} \\ R'_{23} & I & \cdots & R_{3p} \\ \cdot & \cdot & \cdots & \cdot \\ R'_{2p} & R'_{3p} & \cdots & I \end{bmatrix},$$

$R^{*\prime}_{12}$ is the transpose of R^{*}_{12} ,

and the square of the canonical-multiple correlation of set 1 vs. the others is the largest characteristic root of the matrix $R^{*}_{12}\ R^{*-1}_{22}\ R^{*\prime}_{12}$.

Hence $\rho^2_{1.23\ldots p} = Ch_{max}(R^{*}_{12}\ R^{*-1}_{22}\ R^{*\prime}_{12})$ (5.9)

and by the same process we can find all of the squares of the canonical-multiple correlations of one set vs. the others i.e. $\rho^2_{2.13\ldots p}, \quad \rho^2_{3.124\ldots p}, \ \cdots, \quad \rho^2_{p.123\ldots(p-1)}$ (5.10)

Let $v_1 = 1 - \rho^2_{1.23\ldots p}$, $v_2 = 1 - \rho^2_{2.13\ldots p}, \cdots$,

$v_p = 1 - \rho^2_{p.12\ldots(p-1)}$. (5.11)

Let us denote the canonical-partial correlation for i^{th}

and j^{th} sets given the others by

$$u_{ij} = \rho_{ij.k...} ; \tag{5.12}$$

then the inverse of a correlation matrix would be

$$P^{-1} = \begin{bmatrix} \frac{1}{v_1} & \frac{-u_{12}}{\sqrt{v_1 v_2}} & \frac{-u_{13}}{\sqrt{v_1 v_3}} & \cdots & \frac{-u_{1p}}{\sqrt{v_1 v_p}} \\ \frac{-u_{12}}{\sqrt{v_1 v_2}} & \frac{1}{v_2} & \frac{-u_{23}}{\sqrt{v_2 v_3}} & \cdots & \frac{-u_{2p}}{\sqrt{v_2 v_p}} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \frac{-u_{1p}}{\sqrt{v_1 v_p}} & \frac{-u_{2p}}{\sqrt{v_2 v_p}} & \frac{-u_{3p}}{\sqrt{v_3 v_p}} & \cdots & \frac{1}{v_p} \end{bmatrix} \tag{5.13}$$

In matrix form, $P^{-1} = D_{v^{-\frac{1}{2}}} (2I - U) D_{v^{-\frac{1}{2}}}$ (5.14)

where U is the matrix of the canonical-partial correlations with typical element u_{ij}, and $D_{v^{-\frac{1}{2}}}$ is a diagonal matrix with typical element $\frac{1}{\sqrt{v_i}}$.

CHAPTER VI

ILLUSTRATIONS

6.1 Three Categorical Variables, Each With Three States

Let us consider a three-set case, each having three states, for example: color, taste and harvesting region of fruit. The contingency table is then constructed as follow:

(1) The first set (color) : Red, Blue and Yellow.

(2) The second set (taste) : Sweet, Sour and Bitter.

(3) The third set (region) : North, South and Central.

Assume that the contingency table (3 x 3 x 3) obtained from a taste testing experiment is

Table 6.1.1

	N			C			S		
	SW	S	B	SW	S	B	SW	S	B
R	30	70	30	37	28	10	53	5	7
B	10	50	40	15	25	30	35	25	10
Y	75	20	5	27	0	23	30	0	20

There are three two-way tables:

(a) Tastes vs. Colors.

(b) Regions vs. Colors.

(c) Tastes vs. Regions.

Table 6.1.2

Tastes vs. Colors

	SW	S	B	Sub-total
R	120	103	47	270
B	60	100	90	250
Y	132	20	48	200
Sub-total	312	223	185	720

Table 6.1.3

Regions vs. Colors

	N	C	S	Sub-total
R	130	75	65	270
B	100	70	80	250
Y	100	50	50	200
Sub-total	330	195	195	720

Table 6.1.4

Tastes vs. Regions

	N	C	S	Sub-total
SW	115	79	118	312
S	140	53	30	223
B	75	63	47	185
Sub-total	330	195	195	720

From these contingency two-way tables, we find the following E matrices:

$$E_{11} = \begin{bmatrix} 168.7500 & -93.7500 & -75.0000 \\ -93.7500 & 163.1944 & -69.4444 \\ -75.0000 & -69.4444 & 144.4444 \end{bmatrix}$$

$$E_{22} = \begin{bmatrix} 176.8000 & -96.6333 & -80.6666 \\ -96.6333 & 153.9379 & -57.2986 \\ -80.6666 & -57.2986 & 137.4653 \end{bmatrix}$$

$$E_{33} = \begin{bmatrix} 178.7500 & -89.3750 & -89.3750 \\ -89.3750 & 142.1875 & -52.8125 \\ -89.3750 & -52.8125 & 142.1875 \end{bmatrix}$$

$$E_{12} = \begin{bmatrix} 3.0000 & 19.3750 & -22.3750 \\ -48.3333 & 22.5964 & 25.7638 \\ 45.3333 & -41.9444 & -3.3888 \end{bmatrix}$$

$$E_{13} = \begin{bmatrix} 6.2600 & 1.8750 & -8.1250 \\ -14.5833 & 2.2916 & 12.2916 \\ 8.3333 & -4.1666 & -4.1666 \end{bmatrix}$$

$$E_{23} = \begin{bmatrix} -28.0000 & -5.5000 & 33.5000 \\ 37.7916 & -7.3958 & -30.3958 \\ -9.7916 & 12.8958 & -3.1042 \end{bmatrix}$$

Using the Gauss-Doolittle foreward method, we then find $T^{(-1)}$ matrices.

$$T_1^{(-1)} = \begin{bmatrix} 0.076980 & 0.0 & 0.0 \\ 0.052705 & 0.094868 & 0.0 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 0.075207 & 0.0 & 0.0 \\ 0.054355 & 0.099447 & 0.0 \end{bmatrix}$$

$$T_3^{(-1)} = \begin{bmatrix} 0.074958 & 0.0 & 0.0 \\ 0.050673 & 0.101273 & 0.0 \end{bmatrix}$$

and $R = D_{T(-1)} \; E \; D_{T(-1)}'$

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{12}' & R_{22} & R_{23} \\ R'_{13} & R_{23}' & R_{33} \end{bmatrix}$$

$$R_{11} = \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} = R_{22} = R_{33}$$

$$R_{12} = \begin{bmatrix} 0.017368 & 0.160876 \\ -0.322955 & 0.073841 \end{bmatrix}$$

$$R_{13} = \begin{bmatrix} 0.035986 & 0.038980 \\ -0.078842 & -0.021350 \end{bmatrix}$$

$$R_{23} = \begin{bmatrix} -0.157504 & -0.148522 \\ 0.167268 & 0.008479 \end{bmatrix}$$

where

$$D_{T(-1)} = \begin{bmatrix} T_1^{(-1)} & 0 & 0 \\ 0 & T_2^{(-1)} & 0 \\ 0 & 0 & T_3^{(-1)} \end{bmatrix}$$

After initialization of the canonical weights, by the process of the Fletcher-Powell method, the normalized canonical weights for each set turned out to be

$$\underline{a}_1' = [\ 0.0547 \quad 0.998\ ]$$
$$\underline{a}_2' = [\ 0.929 \quad -0.369\ ]$$
$$\underline{a}_3' = [\ 0.856 \quad 0.516\ ]$$

These canonical weights will minimize the log determinant of the canonical correlation matrix, we further calculate the matrix with elements $\underline{a}_i' R_{ij} \underline{a}_j = \rho_{ij}$

$$P = \begin{bmatrix} 1.00000 & -0.33853 & -0.07564 \\ -0.33853 & 1.00000 & -0.25115 \\ -0.07564 & -0.25115 & 1.00000 \end{bmatrix} \qquad (6.1.1)$$

the inverse of P is

$$P^{-1} = \begin{bmatrix} 1.16571 & 0.44484 & 0.19992 \\ 0.44484 & 1.23707 & 0.34435 \\ 0.19992 & 0.34435 & 1.10160 \end{bmatrix} \qquad (6.1.2)$$

and $|P| = 0.80373$, the minimum determinant.

For comparison , an attempt will be made to approximate P^{-1} by the canonical-partial, canonical-multiple correlation described in CHAPTER V. We partition the super-matrix R in the following manner:

$$R^* = \begin{bmatrix} R_{11}^* & R_{12}^* \\ R_{12}^{*\prime} & R_{22}^* \end{bmatrix}$$

$$R_{11}^* = I - R_{13}R_{13}' = \begin{bmatrix} 0.997184 & 0.003669 \\ 0.003669 & 0.993328 \end{bmatrix}$$

$$R_{22}^* = I - R_{23}R_{23}' = \begin{bmatrix} 0.953133 & 0.027605 \\ 0.027605 & 0.971949 \end{bmatrix}$$

$$R_{12}^{*} = R_{12} - R_{13}R_{23}' = \begin{bmatrix} 0.028826 & 0.154526 \\ -0.348544 & 0.087210 \end{bmatrix}$$

$$T_1^{*-1} = \begin{bmatrix} 1.001410 & 0.0 \\ -0.003692 & 1.003350 \end{bmatrix}$$

$$U_{12.3} = T_1^{*-1} R_{12}^{*} R_{22}^{*-1} R_{12}^{*'} T_1^{*-1'}$$

$$= \begin{bmatrix} 0.025269 & 0.004788 \\ 0.004788 & 0.138094 \end{bmatrix}$$

The canonical-partial correlation for set 1 and set 2 given set 3 is the square root of the largest characteristic root of $U_{12.3}$,

$$\rho_{12.3} = \sqrt{Ch_{max}(U_{12.3})} = \sqrt{0.138296} = 0.371882$$

We then continue the process and partition the super-matrix R as follow, in order to find $\rho_{13.2}$,

$$\begin{bmatrix} I & R_{13} & R_{12} \\ R_{13}' & I & R_{23}' \\ R_{12}' & R_{23} & I \end{bmatrix} , \quad R^{*} = \begin{bmatrix} R_{11}^{*} & R_{13}^{*} \\ R_{13}^{*'} & R_{33}^{*} \end{bmatrix}$$

where

$$R_{11}^{*} = I - R_{12}R_{12}' = \begin{bmatrix} 0.973817 & -0.006096 \\ -0.006096 & 0.883687 \end{bmatrix}$$

$$R_{33}^{*} = I - R_{23}'R_{23} = \begin{bmatrix} 0.947213 & -0.024811 \\ -0.024811 & 0.977869 \end{bmatrix}$$

$$R_{13}^{*} = R_{13} - R_{12}R_{23}' = \begin{bmatrix} 0.011812 & 0.040196 \\ -0.143635 & -0.071428 \end{bmatrix}$$

$$T_1^{*-1} = \begin{bmatrix} 1.056430 & 0.0 \\ 0.006659 & 1.063800 \end{bmatrix}$$

$$U_{13.2} = T_1^{*-1} R_{13}^{*} R_{33}^{*-1} R_{13}^{*\prime} T_1^{*-1\prime}$$

$$= \begin{bmatrix} 0.001875 & -0.005278 \\ -0.005378 & 0.311262 \end{bmatrix}$$

The canonical-partial correlation of set 1 and set 3 given set 2 is the square root of the largest characteristic root of $U_{13.2}$,

$$\rho_{13.2} = \sqrt{Ch_{max}(U_{13.2})} = \sqrt{0.032049} = 0.179023$$

For the canonical-partial correlation $\rho_{23.1}$, we proceed as follow:

$$\begin{bmatrix} I & R_{23} & R_{12}' \\ R_{23}' & I & R_{13}' \\ R_{12} & R_{13} & I \end{bmatrix}, \quad R^{*} = \begin{bmatrix} R_{22}^{*} & R_{23}^{*} \\ R_{23}^{*\prime} & R_{33}^{*} \end{bmatrix}$$

where

$$R_{22}^{*} = I - R_{12}'R_{12} = \begin{bmatrix} 0.888838 & 0.021792 \\ 0.021792 & 0.968666 \end{bmatrix}$$

$$R_{33}^{*} = I - R_{13}'R_{13} = \begin{bmatrix} 0.992489 & -0.003086 \\ -0.003086 & 0.998024 \end{bmatrix}$$

$$R_{23}^{*} = R_{23} - R'_{12}R_{13} = \begin{bmatrix} -0.184380 & -0.156307 \\ 0.167301 & 0.003785 \end{bmatrix}$$

$$T_2^{*-1} = \begin{bmatrix} 1.060690 & 0.0 \\ -0.024917 & 1.016320 \end{bmatrix}$$

The canonical-partial correlation of set 2 and set 3

given set 1 is the square root of the largest characteristic root of $U_{23.1}$,

$$U_{23.1} = T_2^{*-1} R_{23}^{*} R_{33}^{*-1} R_{23}^{*\prime} T_2^{*-1}$$

$$= \begin{bmatrix} 0.066282 & -0.035792 \\ -0.035792 & 0.030794 \end{bmatrix}$$

$$\rho_{23.1} = \sqrt{Ch_{max}(U_{23.1})} = \sqrt{0.088487} = 0.297466$$

Hence the canonical-partial correlation matrix is

$$\begin{bmatrix} 1.0 & 0.37188 & 0.17902 \\ 0.37188 & 1.0 & 0.29747 \\ 0.17902 & 0.29747 & 1.0 \end{bmatrix}$$

To find the canonical-multiple correlations from the super-matrix R, we will partition the R into the following form

$$R = \begin{bmatrix} I & R_{12} & R_{13} \\ R_{12}' & I & R_{23} \\ R_{13}' & R_{23}' & I \end{bmatrix} = \begin{bmatrix} I & R_{12}^{*} \\ R_{12}^{*\prime} & R_{22}^{*} \end{bmatrix}$$

where $R_{12}^{*} = [R_{12} \quad R_{13}]$ and $R_{22}^{*} = \begin{bmatrix} I & R_{23} \\ R_{23}' & I \end{bmatrix}$

and $V_{1.23} = R_{12}^{*} R_{22}^{*-1} R_{12}^{*\prime}$

$$= \begin{bmatrix} 0.028009 & 0.001189 \\ 0.001189 & 0.143878 \end{bmatrix}$$

The canonical-multiple correlation of set 1 vs the others is the square root of the largest characteristic root of $V_{1.23}$,

$$\rho_{1.23} = \sqrt{Ch_{max}(V_{1.23})} = \sqrt{0.143890} = 0.379328$$

and the associated characteristic vector is

$$[\ 0.010257 \quad 0.999947\ ]$$

The canonical-multiple correlation of set 2 vs the others is obtained by partitioning the super-matrix R in the following manner:

$$R = \begin{bmatrix} I & R'_{12} & R_{23} \\ R_{12} & I & R_{13} \\ R'_{23} & R'_{13} & I \end{bmatrix} = \begin{bmatrix} I & R^*_{23} \\ R^{*\prime}_{23} & 1 \end{bmatrix}$$

where $R^*_{23} = [\ R'_{12} \quad R_{23}]$ abd $R^*_{33} = \begin{bmatrix} I & R_{13} \\ R'_{13} & I \end{bmatrix}$

$$V_{2.13} = R^*_{23}\, R^{*-1}_{33}\, R^{*\prime}_{23}$$

$$= \begin{bmatrix} 0.170075 & -0.053549 \\ -0.053549 & 0.059554 \end{bmatrix}$$

The canonical-multiple correlation of set 2 vs the others is the square root of the largest characteristic root of $V_{2,13}$,

$$\rho_{2.13} = \sqrt{Ch_{max}(V_{2.13})} = \sqrt{0.191763} = 0.437908$$

and the associated characteristic vector is

$$[\ 0.926861 \ -0.375404\ ]$$

The canonical-multiple correlation of set 3 vs the others is obtained by partitioning the super-matrix R in the following manner:

$$R = \begin{bmatrix} I & R_{13}' & R_{23}' \\ R_{13} & I & R_{12} \\ R_{23} & R_{12}' & I \end{bmatrix} = \begin{bmatrix} I & R_{13}^{*} \\ R_{13}^{*\prime} & R_{11}^{*} \end{bmatrix}$$

where

$$R_{13}^{*} = \begin{bmatrix} R_{13}' & R_{23}' \end{bmatrix} \text{ and } R_{11}^{*} = \begin{bmatrix} I & R_{12} \\ R_{12}' & I \end{bmatrix} ,$$

$$V_{3.12} = R_{13}^{*} R_{11}^{*-1} R_{13}^{*\prime}$$

$$= \begin{bmatrix} 0.076253 & 0.036862 \\ 0.036862 & 0.029523 \end{bmatrix}$$

The canonical-multiple correlation of set 3 vs the others is the square root of the largest characteristic root of $V_{3.12}$.

$$\rho_{3.12} = \sqrt{Ch_{max}(V_{3.12})} = \sqrt{0.096532} = 0.310695$$

and the associated characteristic vector is

$$[\ 0.876174 \quad 0.481994\ ]$$

In accordance with the discussion in CHAPTER V,

$$\rho^{ij} = \frac{-\ \rho_{ij.k}}{\sqrt{1 - \rho^2_{i.jk}}\sqrt{1 - \rho^2_{j.ik}}} \qquad \text{for } i \neq j$$

at this stage, the sign of the partial correlation become important. A canonical-partial correlation is defined to be positive but the entries in the inverse of a correlation matrix can be positive or negative. One technique of choosing signs is to use those which produce an inverse whose diagonal elements are near unity. In the present example,

assuming all partial correlations to be negative we obtained an inverse whose diagonal elements were close to one, namely:

$$P^{-1} = \begin{bmatrix} 1.16807 & 0.44707 & 0.20356 \\ 0.44707 & 1.12373 & 0.34811 \\ 0.20356 & 0.34811 & 1.10684 \end{bmatrix}$$

and

$$P=(P^{-1})^{-1}= \begin{bmatrix} 0.99946 & -0.33946 & -0.07705 \\ -0.33946 & 1.00199 & -0.25270 \\ -0.07705 & -0.25270 & 0.99711 \end{bmatrix}$$

This is quite similar to the correlation matrix (6.1.1) and (6.1.2) based upon the minimum-determinant solution.

Three principally different approaches (canonical-multiple initialization, minimum-determinant and construction of the inverse of a correlation matrix) lead to very similar results. This fact is additional evidence of the existence of optimumscale values derived from the data, quite independent of the method of analysis.

The reduction to the original categorical weights for the E matrix is as follow:

$$\underline{a}_1' = [\ 0.0547 \quad 0.998\ ]$$

$$\underline{a}_2' = [\ 0.929 \quad -0.369\ ]$$

$$\underline{a}_3' = [\ 0.856 \quad 0.516\ ]$$

The marginal totals for each set are

$$\underline{n}_1' = [\ 270 \quad 250 \quad 200\ ]$$

$$\underline{n}_2' = [\ 312 \quad 223 \quad 185\ ]$$

$$\underline{n}_3' = [\ 300 \quad 195 \quad 195\ ]$$

The grand total is: $n = 720$

Reduction to the original categorical weights results are

$$\underline{w}_1' = [0.07054 \quad 1.08735 \quad -1,45442]$$

$$\underline{w}_2' = [1.06254 \quad -1.25980 \quad -0.27338]$$

$$\underline{w}_3' = [0.93106 \quad -0.08640 \quad -1.48923]$$

6.2 Three Categorical Variables, with 2, 5 and 2 States

Example of a contingency table:

Table 6.2.1

	Men					Women				
	A	B	C	D	E	A	B	C	D	E
Survive	741	742	345	188	79	896	718	276	93	35
Death	360	297	150	75	79	420	334	175	59	41
Sub-total	1101	1039	495	263	158	1316	1052	451	152	76

Remark: From Cramér's "Mathematical Methods of Statistics" pp 450, table 30.7.1.

(1) The first set (sex) : Men, women.

(2) The second set (age) : A - 15 to 24 year-old group.
B - 25 to 34 year-old group.
C - 35 to 44 year-old group.
D - 45 to 54 year-old group.
E - 55 year-old and above.

(3) The third set (risk) : Survive, death.

Of course, categorical scaling is required for the five-

state set only since weights are arbitrary for the two-level factors. We will, however, proceed formally as in the previous case.

There are three two-way tables:

Table 6.2.2

Sex vs. Age

	Men	Women	Sub-total
A	1101	1316	2417
B	1039	1052	2[illegible]9[illegible]
C	495	451	[illegible]46
D	263	152	415
E	158	76	234
Sub-total	3056	3047	6103

Table 6.2.3

Sex vs. Risk

	Men	Women	Sub-total
Survive	2095	2018	4113
Death	961	1029	1990
Sub-total	3056	3047	6103

Table 6.2.4

Age vs. Risk

	A	B	C	D	E	Sub-total
Survive	1637	1460	621	281	114	4113
Death	780	631	325	134	120	1990
Sub-total	2417	2091	946	415	234	6103

The E matrices are as follow:

$$E_{11}=\begin{bmatrix} 1525.7467 & -1525.7467 \\ -1525.7467 & 1525.7467 \end{bmatrix}$$

$$E_{22}=\begin{bmatrix} 1459.7840 & -828.1086 & -374.6489 & -164.3544 & -92.6721 \\ -828.1086 & 1374.5849 & -324.1170 & -142.1866 & -80.1727 \\ -374.6489 & -324.1170 & 799.3646 & -64.3274 & -36.2713 \\ -164.3544 & -142.1866 & -64.3274 & 386.7803 & -15.9118 \\ -92.6721 & -80.1727 & -36.2713 & -15.9118 & 255.0280 \end{bmatrix}$$

$$E_{33}=\begin{bmatrix} 1341.1224 & -1341.1224 \\ -1341.1224 & 1341.1224 \end{bmatrix}$$

$$E_{12}=\begin{bmatrix} -109.2822 & -8.0418 & 21.3025 & 55.1940 & 40.8275 \\ 109.2822 & 8.0418 & -21.3025 & -55.1940 & -40.8275 \end{bmatrix}$$

$$E_{13}=\begin{bmatrix} 35.4673 & -35.4673 \\ -35.4673 & 35.4673 \end{bmatrix}$$

$$E_{23}=\begin{bmatrix} 8.1091 & -8.1091 \\ 50.8106 & -50.8106 \\ -16.5386 & 16.5386 \\ 1.3187 & -1.3187 \\ -43.6998 & 43.6998 \end{bmatrix}$$

The T conditional inverses are:

$$T_1^{(-1)} = [0.025601 \quad 0.0]$$

$$T_2^{(-1)} = \begin{bmatrix} 0.026173 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.018859 & 0.033244 & 0.0 & 0.0 & 0.0 \\ 0.030230 & 0.030230 & 0.050969 & 0.0 & 0.0 \\ 0.052275 & 0.052275 & 0.052275 & 0.081750 & 0.0 \end{bmatrix}$$

$$T_3^{(-1)} = 0.027306 \quad 0.0$$

The super-matrix R is

$$R_{11} = 1$$

$$R_{12} = [-0.073226 \ -0.059607 \ -0.063003 \ -0.012989]$$

$$R_{22} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{13} = 0.024794$$

$$R_{23} = \begin{bmatrix} 0.005796 \\ 0.050301 \\ 0.025619 \\ 0.063440 \end{bmatrix}$$

$$R_{33} = 1$$

By initialization of the canonical weights and by the Fletcher-Powell method, we found the normalized categorical weights for each set are

$$a_1 = 1 = a_3$$

$$\underline{a}_2' = [\,0.501 \quad 0.598 \quad 0.517 \quad 0.349\,]$$

Further, we calculate the canonical correlations with (i, j) element $\underline{a}_i' R_{ij} \underline{a}_j$

$$P = \begin{bmatrix} 1.00000 & -0.10956 & 0.02479 \\ -0.10956 & 1.00000 & -0.06845 \\ 0.02479 & -0.06845 & 1.00000 \end{bmatrix} . \tag{6.2.1}$$

Here, the (1,3) and (3.1) elements are positive; if we want **all** off-diagonal elements to be negatives we can make the signs in $\underline{a}_2'$ negative, which changes the sign in the (1,2) and (2,3) elements.

The inverse of P is

$$P^{-1} = \begin{bmatrix} 1.01322 & 0.11326 & 0.03287 \\ 0.11326 & 1.01736 & 0.07245 \\ 0.03287 & 0.07245 & 1.00577 \end{bmatrix} \tag{6.2.2}$$

and $|P| = 0.982324$

Again, to approximate the matrix P^{-1} by canonical-partial and canonical-multiple correlations we will make the usual partitions of the super-matrix R. First, the canonical-partial correlation of set 1 and set 2 given set 3 is obtained from

$$R = \begin{bmatrix} I & R_{12} & R_{13} \\ R_{12}' & I & R_{23} \\ R_{13}' & R_{23}' & I \end{bmatrix} .$$

and $R^{*}_{11} = I - R_{13} R_{13}' = 0.999385$

$R^{*}_{22} = I - R_{23} R_{23}'$

$$= \begin{bmatrix} 0.999966 & -0.000292 & -0.000149 & -0.000368 \\ -0.000292 & 0.997469 & -0.001289 & -0.003191 \\ -0.000149 & -0.001289 & 0.999343 & -0.001625 \\ -0.000368 & -0.003191 & -0.001625 & 0.995975 \end{bmatrix}$$

$R^*_{12} = R_{12} - R_{13} R'_{23}$

$= [-0.073369 \ -0.060854 \ -0.063638 \ -0.014562]$

Hence, the canonical-partial correlation of set 1 and set 3 given set 3 is the square root of the largest characteristc root of

$$U_{12.3} = R^{*-1}_{11} R^*_{12} R^{*-1}_{22} R^{*'}_{12} = 0.013393$$

and

$$\rho_{12.3} = \sqrt{Ch_{max}(U_{12.3})} = \sqrt{0.013393} = 0.115728$$

For the canonical-partial correlation of set 1 and set 3 given set 2, we have

$R^*_{11} = I - R_{12} R'_{12} = 0.986946$

$R^*_{33} = I - R'_{23} R_{23} = 0.992755$

$R^*_{13} = R_{13} - R_{12} R'_{23} = 0.030655$

$U_{13.2} = R^{*-1}_{11} R^*_{13} R^{*-1}_{33} R^{*'}_{13} = 0.000959$

Therefore the canonical-partial correlation of set 1 and set 3 given set 2 would be the square root of the largest characteristic root of $U_{13.2}$, hence

$\rho_{13.2} = 0.030969$

Finally, the canonical-partial correlation of set 2 and set 3 given set 1 is

$$R^*_{22} = I - R'_{12}\,R_{12}$$

$$= \begin{bmatrix} 0.994638 & -0.004365 & -0.004614 & -0.000951 \\ -0.004365 & 0.996447 & -0.003755 & -0.000774 \\ -0.004614 & -0.003755 & 0.996030 & -0.000818 \\ -0.000951 & -0.000774 & -0.000818 & 0.999831 \end{bmatrix}$$

$$R^*_{33} = I - R'_{13}\,R_{13} = 0.999385$$

$$R^*_{23} = R_{23} - R'_{12}\,R_{13} = \begin{bmatrix} 0.007611 \\ 0.051779 \\ 0.027181 \\ 0.063763 \end{bmatrix}$$

The canonical-partial correlation of set 2 and set 3 given set 1 is the square root of the largest characteristic root of $U_{23.1}$, where

$$U_{23.1} = \begin{bmatrix} 0.000058 & 0.000396 & 0.000209 & 0.000488 \\ 0.000396 & 0.002696 & 0.001427 & 0.003316 \\ 0.000209 & 0.001427 & 0.000755 & 0.001755 \\ 0.000488 & 0.003316 & 0.001755 & 0.004078 \end{bmatrix}$$

$$\rho_{23.1} = 0.087103$$

So the canonical-partial correlation matrix is

$$\begin{bmatrix} 1.00000 & 0.11573 & 0.03097 \\ 0.11573 & 1.00000 & 0.08710 \\ 0.03097 & 0.08710 & 1.00000 \end{bmatrix}$$

For the canonical-multiple correlations we will partition the super-matrix R as described earlier; for the canonical-multiple correlation of set 1 vs. the others:

$$R^*_{12} = [R_{12} \quad R_{13}]$$

$$R^*_{22} = \begin{bmatrix} I & R_{23} \\ R'_{23} & I \end{bmatrix}$$

and $V_{1.23} = R^*_{12} R^{*-1}_{22} R^{*'}_{12} = 0.013999$

The square root of the largest characteristic root of $V_{1.23}$ is the canonical-multiple correlation of set 1 vs the others:

$$\rho_{1.23} = 0.118320$$

and the associated characteristic vector is, of course

$$[1]$$

For the canonical-multiple correlation of set 2 vs the others, we have

$$R^*_{23} = [R'_{12} \quad R_{23}] \text{ and } R^*_{33} = \begin{bmatrix} I & R_{13} \\ R'_{13} & I \end{bmatrix}$$

$$V_{2.13} = R^*_{23} R^{*-1}_{33} R^{*'}_{23}$$

$$= \begin{bmatrix} 0.005419 & 0.004759 & 0.004821 & 0.001437 \\ 0.004759 & 0.006236 & 0.005164 & 0.004078 \\ 0.004821 & 0.005164 & 0.004709 & 0.002553 \\ 0.001437 & 0.004077 & 0.002553 & 0.004237 \end{bmatrix}$$

The square root of the largest characteristic root of $V_{2.13}$ is the canonical-multiple correlation of set 2 vs the others:

$$\rho_{2.13} = 0.13068$$

and the associated characteristic vector is

$$[0.501723 \quad 0.598517 \quad 0.517728 \quad 0.349290]$$

Finally, for the canonical -multiple correlation of set 3 vs the others, we have

$$R^*_{13} = [R'_{13} \quad R'_{23}] \text{ and } R^*_{11} = \begin{bmatrix} I & R_{12} \\ R'_{12} & I \end{bmatrix}$$

$$V_{3.12} = R^*_{13} R^{*-1}_{11} R^{*\prime}_{13} = 0.008197$$

Then the canonical-multiple correlation of set 3 vs the others is the square root of $V_{3.12}$, hence

$$\rho_{3.12} = 0.090537$$

and the associated characteristic vector is

$$[1]$$

So the inverse of a canonical correlation matrix P would be

$$P^{-1} = \begin{bmatrix} 1.01419 & 0.11756 & 0.03132 \\ 0.11756 & 1.01736 & 0.08822 \\ 0.03132 & 0.08822 & 1.00826 \end{bmatrix}$$

and

$$P=(P^{-1})^{-1} = \begin{bmatrix} 0.99983 & -0.11369 & -0.02110 \\ -0.11369 & 1.00337 & -0.08426 \\ -0.02110 & -0.08426 & 0.99983 \end{bmatrix}$$

and $|P| = 0.982158$, the minimum determinant.

Again, when compare with (6.2.1) and (6.2.2) the approximation is quite close.

For the reduction of the categorical weights for E matrix, we have from Fletcher-Powell method the normalized canonical weights

$a_1 = 1 = a_3$

$\underline{a}_2' = [0.501 \quad 0.598 \quad 0.517 \quad 0.349]$

The marginal totals for each set are

$\underline{n}_1' = [3956 \quad 3047]$

$\underline{n}_2' = [2417 \quad 2091 \quad 946 \quad 415 \quad 234]$

$\underline{n}_3' = [4113 \quad 1990]$

with the grand total n = 6103.

The final categorical weights for the E matrix are

$\underline{w}_1' = [0.99853 \; -1.00147]$

$\underline{w}_2' = [0.61940 \quad 0.26653 \; -0.44928 \; -1.70580 \; -3.93789]$

$\underline{w}_3' = [0.69558 \; -1.43764]$.

6.3 <u>Four Categorical Variables, with 4, 4, 3 and 3 States</u>

Let us consider a case of four sets with the first and the second sets having four states, the third and the fourth sets having three states (4 x 4 x 3 x 3).

There will be six two-way tables; the calculation process is the same as two previous cases (6.1 and 6.2).

The contingency table is

Table 6.3.1

		A_1				A_2				A_3				A_4			
		B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4
C_1	D_1	2	12	2	8	2	10	100	150	1	6	80	80	5	1	1	1
	D_2	10	15	4	1	5	6	2	40	4	8	60	2	20	10	2	5
	D_3	1	10	2	1	2	1	2	10	1	1	8	1	2	1	1	1
C_2	D_1	2	8	1	1	1	1	1	2	1	4	10	1	10	4	2	1
	D_2	50	40	10	1	1	2	4	5	5	10	12	1	40	50	10	1
	D_3	120	80	8	2	1	1	1	12	9	6	5	1	150	80	10	[illegible]
C_3	D_1	20	20	1	4	5	6	3	60	8	4	40	2	15	10	5	2
	D_2	40	200	10	4	1	2	6	20	10	40	50	5	100	50	10	2
	D_3	30	60	4	1	5	4	3	15	8	6	20	2	40	20	5	2

The six two-way tables:

Table 6.3.2

A vs B

	A_1	A_2	A_3	A_4	Sub-total
B_1	275	23	47	382	727
B_2	445	33	85	226	789
B_3	42	122	285	46	495
B_4	22	314	95	16	447
Sub-total	784	492	512	670	2458

Table 6.3.3

A vs C

	A_1	A_2	A_3	A_4	Sub-total
C_1	68	330	252	50	700
C_2	323	32	65	359	779
C_3	393	130	195	261	979
Sub-total	784	492	512	670	2458

Table 6.3.4

A vs D

	A_1	A_2	A_3	A_4	Sub-total
D_1	81	341	237	57	716
D_2	384	94	207	300	985
D_3	319	57	68	313	757
Sub-total	784	492	512	670	2458

Table 6.3.5

B vs C

	B_1	B_2	B_3	B_4	Sub-total
C_1	55	81	264	300	700
C_2	390	286	74	29	779
C_3	282	422	157	118	979
Sub-total	727	789	495	447	2458

Table 6.3.6

B vs D

	B_1	B_2	B_3	B_4	Sub-total
D_1	72	86	246	312	716
D_2	286	433	180	86	985
D_3	369	170	69	49	757
Sub-total	727	789	495	447	2458

Table 6.3.7

C vs D

	C_1	C_2	C_3	Sub-total
D_1	461	50	205	716
D_2	194	242	549	985
D_3	45	487	225	757
Sub-total	700	779	979	2458

The E matrix are as follow:

$$E_{11} = \begin{bmatrix} 533.936 & -156.927 & -163.306 & -213.702 \\ -156.927 & 393.519 & -102.483 & -134.109 \\ -163.306 & -102.483 & 405.350 & -139.560 \\ -213.702 & -134.109 & -139.560 & 487.371 \end{bmatrix}$$

$$E_{22} = \begin{bmatrix} 511.976 & -233.361 & -146.404 & -132.208 \\ -233.361 & 535.736 & -158.891 & -143.483 \\ -146.405 & -158.891 & 395.315 & -90.018 \\ -132.208 & -143.483 & -90.018 & 365.710 \end{bmatrix}$$

$$E_{33} = \begin{bmatrix} 500.650 & -221.847 & -278.803 \\ -221.847 & 532.115 & -310.268 \\ -278.803 & -310.268 & 589.072 \end{bmatrix}$$

$$E_{44} = \begin{bmatrix} 507.433 & -286.924 & -220.509 \\ -286.924 & 590.278 & -303.354 \\ -220.509 & -303.354 & 523.863 \end{bmatrix}$$

$$E_{12} = \begin{bmatrix} 43.117 & 193.341 & -115.884 & -120.574 \\ -122.518 & -124.928 & 22.919 & 224.527 \\ -104.433 & -79.348 & 181.891 & 1.890 \\ 183.834 & 10.935 & -88.927 & -105.842 \end{bmatrix}$$

$$E_{13} = \begin{bmatrix} -155.270 & 74.531 & 80.739 \\ 189.886 & -123.926 & -65.959 \\ 106.190 & -97.265 & -8.925 \\ -140.805 & 146.660 & -5.855 \end{bmatrix}$$

$$E_{23} = \begin{bmatrix} -152.038 & 159.516 & -7.558 \\ -143.694 & 35.947 & 107.748 \\ 123.031 & -82.878 & -40.154 \\ 172.701 & -112.665 & -60.036 \end{bmatrix}$$

$$E_{14} = \begin{bmatrix} -147.374 & 69.826 & 77.548 \\ 197.683 & -103.160 & -94.523 \\ 87.858 & 1.825 & -89.683 \\ -138.166 & 31.509 & 106.657 \end{bmatrix}$$

$$E_{24} = \begin{bmatrix} -139.770 & -5.332 & 145.102 \\ -143.830 & 116.822 & 72.009 \\ 101.809 & -18.362 & -83.447 \\ 181.791 & -93.127 & -88.664 \end{bmatrix}$$

$$E_{34} = \begin{bmatrix} 257.094 & -86.513 & -170.581 \\ -176.917 & -70.170 & 247.088 \\ -80.177 & 156.683 & -76.507 \end{bmatrix}$$

The T conditional inverses are

$$T_1^{(-1)} = \begin{bmatrix} 0.043279 & 0.0 & 0.0 & 0.0 \\ 0.015769 & 0.053652 & 0.0 & 0.0 \\ 0.025427 & 0.025427 & 0.058699 & 0.0 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 0.044195 & 0.0 & 0.0 & 0.0 \\ 0.021997 & 0.048259 & 0.0 & 0.0 \\ 0.034287 & 0.034287 & 0.065248 & 0.0 \end{bmatrix}$$

$$T_3^{(-1)} = \begin{bmatrix} 0.044692 & 0.0 & 0.0 \\ 0.021275 & 0.048012 & 0.0 \end{bmatrix}$$

$$T_4^{(-1)} = \begin{bmatrix} 0.044393 & 0.0 & 0.0 \\ 0.027330 & 0.048335 & 0.0 \end{bmatrix}$$

Hence the super-matrix R consists of

$$R_{11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = R_{22}$$

$$R_{33} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = R_{44}$$

$$R_{12} = \begin{bmatrix} 0.082467 & 0.444844 & 0.023633 \\ -0.260426 & -0.305975 & -0.306343 \\ -0.360152 & -0.322009 & 0.163962 \end{bmatrix}$$

$$R_{13} = \begin{bmatrix} -0.300315 & 0.011902 \\ 0.345889 & -0.098147 \\ 0.317918 & -0.183083 \end{bmatrix}$$

$$R_{23} = \begin{bmatrix} -0.300303 & 0.195692 \\ -0.459394 & 0.033157 \\ -0.094392 & 0.017331 \end{bmatrix}$$

$$R_{14} = \begin{bmatrix} -0.283130 & -0.028251 \\ 0.367669 & 0.012056 \\ 0.285728 & 0.140120 \end{bmatrix}$$

$$R_{24} = \begin{bmatrix} -0.444625 & -0.006904 \\ -0.136764 & 0.042654 \end{bmatrix}$$

$$R_{34} = \begin{bmatrix} 0.510076 & 0.127246 \\ -0.134265 & -0.334463 \end{bmatrix}$$

By some initialization of the canonical weights, and the Fletcher Powell method we obtain a set of normalized canonical weights which will minimize the log determinant of the canonical correlation matrix,

$$\underline{a}_1' = [-0.531 \quad 0.619 \quad 0.577]$$

$$\underline{a}_2' = [\ 0.561 \quad 0.804 \quad 0.192]$$

$$\underline{a}_3' = [\ 0.946 \quad -0.321]$$

$$\underline{a}_4' = [\ 0.958 \quad 0.285]$$

and P with element $\underline{a}_i' R_{ij} \underline{a}_j$ equals

$$\begin{bmatrix} 1.00000 & -0.75165 & 0.58355 & 0.55026 \\ -0.75165 & 1.00000 & -0.57193 & -0.54388 \\ 0.58355 & -0.57193 & 1.00000 & 0.56931 \\ 0.55026 & -0.54388 & 0.56931 & 1.00000 \end{bmatrix} \quad (6.3.1)$$

By taking negative signs of $\underline{a}_2$, we can make the off-diagonal elements all positive, the inverse of P is then

$$P^{-1} = \begin{bmatrix} 2.57275 & -1.48402 & -0.45289 & -0.35071 \\ -1.48402 & 2.51606 & -0.38298 & -0.33380 \\ -0.45289 & -0.38298 & 1.80926 & -0.57252 \\ -0.35071 & -0.33380 & -0.57252 & 1.70047 \end{bmatrix} \quad (6.3.2)$$

and $|P| = 0.158256$

We now calculate the canonical-partial correlations:

For $\rho_{12.34}$:

$$R^*_{11} = I - \begin{bmatrix} R_{13} & R_{14} \end{bmatrix} \begin{bmatrix} I & R_{34} \\ R'_{34} & I \end{bmatrix}^{-1} \begin{bmatrix} R'_{13} \\ R'_{14} \end{bmatrix}$$

$$= \begin{bmatrix} 0.886809 & 0.136459 & 0.115606 \\ 0.136459 & 0.825142 & -0.149995 \\ 0.115606 & -0.149995 & 0.848485 \end{bmatrix}$$

$$R^*_{12} = R_{12} - \begin{bmatrix} R_{13} & R_{14} \end{bmatrix} \begin{bmatrix} I & R_{34} \\ R'_{34} & I \end{bmatrix}^{-1} \begin{bmatrix} R'_{23} \\ R'_{24} \end{bmatrix}$$

$$= \begin{bmatrix} -0.027496 & 0.270088 & -0.019955 \\ -0.118013 & -0.091662 & -0.308620 \\ -0.210162 & -0.141697 & 0.205016 \end{bmatrix}$$

$$R^*_{22} = I - \begin{bmatrix} R_{23} & R_{24} \end{bmatrix} \begin{bmatrix} I & R_{34} \\ R'_{34} & I \end{bmatrix}^{-1} \begin{bmatrix} R'_{23} \\ R'_{24} \end{bmatrix}$$

$$= \begin{bmatrix} 0.849399 & -0.168143 & -0.038132 \\ -0.168143 & 0.728007 & -0.070279 \\ -0.038032 & -0.070279 & 0.977854 \end{bmatrix}$$

$$T_1^{(-1)} = \begin{bmatrix} 1.061900 & 0.0 & 0.0 \\ -0.171596 & 1.115140 & 0.0 \\ -0.181826 & 0.233510 & 1.119140 \end{bmatrix}$$

and $R_{11}^{*-1} = T_1^{(-1)'} T_1^{(-1)}$

$$U_{12.34} = T_1^{(-1)} R_{12}^{*} R_{22}^{*-1} R_{12}^{*'} T_1^{(-1)'}$$

$$= \begin{bmatrix} 0.114767 & -0.069291 & -0.103357 \\ -0.069291 & 0.200738 & 0.063874 \\ -0.103357 & 0.063874 & 0.205583 \end{bmatrix}$$

The canonical-partial correlation of set 1 and set 2 given the others is the square root of the largest characteristic root of $U_{12.34}$, hence

$$\rho_{12.34} = 0.579441$$

For $\rho_{13.24}$,

$$R_{11}^{*} = I - \begin{bmatrix} R_{12} & R_{14} \end{bmatrix} \begin{bmatrix} I & R_{24} \\ R_{24}' & I \end{bmatrix}^{-1} \begin{bmatrix} R_{12}' \\ R_{14}' \end{bmatrix}$$

$$= \begin{bmatrix} 0.789515 & 0.175271 & 0.174976 \\ 0.175271 & 0.678013 & -0.141414 \\ 0.174976 & -0.141414 & 0.729794 \end{bmatrix}$$

$$T_1^{(-1)} = \begin{bmatrix} 1.125430 & 0.0 & 0.0 \\ -0.275758 & 1.242160 & 0.0 \\ -0.353969 & 0.347428 & 1.249140 \end{bmatrix}$$

$$R_{13}^{*} = R_{13} - \begin{bmatrix} R_{12} & R_{14} \end{bmatrix} \begin{bmatrix} I & R_{24} \\ R_{24}' & I \end{bmatrix}^{-1} \begin{bmatrix} R_{23} \\ R_{34}' \end{bmatrix}$$

$$= \begin{bmatrix} -0.049887 & -0.029540 \\ 0.061091 & -0.025272 \\ 0.051780 & -0.076573 \end{bmatrix}$$

$$R^*_{33} = I - \begin{bmatrix} R'_{13} & R_{34} \end{bmatrix} \begin{bmatrix} I & R_{24} \\ R'_{24} & I \end{bmatrix}^{-1} \begin{bmatrix} R_{23} \\ R'_{34} \end{bmatrix}$$

$$= \begin{bmatrix} 0.619281 & 0.122208 \\ 0.122208 & 0.858676 \end{bmatrix}$$

$$U_{13.24} = \begin{bmatrix} 0.005679 & -0.007039 & -0.006350 \\ -0.007039 & 0.013984 & 0.020603 \\ -0.006350 & 0.020603 & 0.033003 \end{bmatrix}$$

The canonical-partial correlation of set 1 and set 3 given the others is the square root of the largest characteristic root of $U_{13.24}$, hence

$$\rho_{13.24} = 0.220117$$

For $\rho_{23.14}$,

$$R^*_{22} = \begin{bmatrix} 0.774930 & -0.234182 & -0.038919 \\ -0.234182 & 0.585180 & -0.078306 \\ -0.038297 & -0.078306 & 0.835689 \end{bmatrix}$$

$$R^*_{23} = \begin{bmatrix} -0.044353 & -0.059261 \\ -0.088416 & -0.053181 \\ -0.001943 & 0.015324 \end{bmatrix}$$

$$R^*_{33} = \begin{bmatrix} 0.620651 & 0.135421 \\ 0.135421 & 0.855330 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 1.135970 & 0.0 & 0.0 \\ 0.421434 & 1.394260 & 0.0 \\ 0.114051 & 0.193599 & 1.105690 \end{bmatrix}$$

$$U_{23.14} = \begin{bmatrix} 0.011512 & 0.009798 & 0.003749 \\ 0.009798 & 0.032873 & 0.005151 \\ 0.003749 & 0.005151 & 0.001377 \end{bmatrix}$$

and $\rho_{23.14} = \sqrt{Ch_{max}(U_{23.14})} = \sqrt{0.037733} = 0.194250$

For $\rho_{14.23}$:

$$R^*_{11} = \begin{bmatrix} 0.787713 & 0.174990 & 0.174636 \\ 0.174990 & 0.690170 & -0.174382 \\ 0.174636 & -0.174382 & 0.718933 \end{bmatrix}$$

$$R^*_{14} = \begin{bmatrix} -0.039524 & -0.007767 \\ 0.078144 & -0.004448 \\ 0.032788 & 0.021033 \end{bmatrix}$$

$$R^*_{44} = \begin{bmatrix} 0.636267 & -0.097197 \\ -0.097197 & 0.858221 \end{bmatrix}$$

$$T_1^{(-1)} = \begin{bmatrix} 1.126710 & 0.0 & 0.0 \\ -0.275169 & 1.230110 & 0.0 \\ -0.360182 & 0.361008 & 1.262890 \end{bmatrix}$$

$$U_{14.23} = \begin{bmatrix} 0.003404 & -0.006865 & -0.006351 \\ -0.006865 & 0.019815 & 0.013064 \\ -0.006351 & -0.013064 & 0.011859 \end{bmatrix}$$

$$\rho_{14.23} = \sqrt{Ch_{max}(U_{14.23})} = \sqrt{0.032470} = 0.180195$$

For $\rho_{24.13}$'

$$R^*_{22} = \begin{bmatrix} 0.774793 & -0.238515 & -0.041468 \\ -0.238515 & 0.583674 & -0.071973 \\ -0.041468 & -0.071973 & 0.838285 \end{bmatrix}$$

$$R^*_{24} = \begin{bmatrix} -0.022346 & -0.079087 \\ -0.082556 & 0.051299 \\ -0.037265 & -0.030412 \end{bmatrix}$$

$$R^*_{44} = \begin{bmatrix} 0.635430 & -0.097427 \\ -0.097427 & 0.865801 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 1.136070 & 0.0 & 0.0 \\ 0.430962 & 1.399930 & 0.0 \\ 0.115421 & 0.183174 & 1.102970 \end{bmatrix}$$

$$U_{24.13} = \begin{bmatrix} 0.011342 & 0.002960 & -0.000382 \\ 0.002960 & 0.025073 & 0.012127 \\ -0.000382 & 0.012127 & 0.006164 \end{bmatrix}$$

$$\rho_{24.13} = \sqrt{Ch_{max}(U_{24.13})} = \sqrt{0.031308} = 0.176941$$

For $\rho_{34.12}$'

$$R^*_{33} = \begin{bmatrix} 0.645318 & 0.097515 \\ 0.097515 & 0.941411 \end{bmatrix}$$

$$R^*_{34} = \begin{bmatrix} 0.166011 & 0.064476 \\ -0.042258 & -0.290438 \end{bmatrix}$$

$$R^*_{44} = \begin{bmatrix} 0.663253 & -0.054594 \\ -0.054594 & 0.958841 \end{bmatrix}$$

$$T_3^{(-1)} = \begin{bmatrix} 1.244830 & 0 \\ -0.156976 & 1.038810 \end{bmatrix}$$

$$U_{34.12} = \begin{bmatrix} 0.074305 & -0.054168 \\ -0.054168 & 0.113066 \end{bmatrix}$$

$$\rho_{34.12} = \sqrt{Ch_{max}(U_{34.12})} = \sqrt{0.151216} = 0.388865$$

Hence the canonical-partial correlation matrix is

$$\begin{bmatrix} 1.00000 & 0.57944 & 0.22012 & 0.18019 \\ 0.57944 & 1.00000 & 0.19425 & 0.17694 \\ 0.22012 & 0.19425 & 1.00000 & 0.38886 \end{bmatrix}$$

Now we want to find all of the canonical-multiple correlations of one set vs the others.

For $\rho_{1.234}$,

$$R^*_{12} = \begin{bmatrix} R_{12} & R_{13} & R_{14} \end{bmatrix}$$

$$R^*_{22} = \begin{bmatrix} I & R_{23} & R_{24} \\ R'_{23} & I & R_{34} \\ R'_{24} & R'_{34} & I \end{bmatrix} \qquad \text{and}$$

$$V_{1.234} = R^*_{12}\, R^{*-1}_{22}\, R^{*\prime}_{12}$$

$$= \begin{bmatrix} 0.214976 & -0.179312 & -0.177099 \\ -0.179312 & 0.320683 & 0.150404 \\ -0.177099 & 0.150404 & 0.283560 \end{bmatrix}$$

The canonical-multiple correlation $\rho_{1.234}$ is the square root of the largest characteristic root of $V_{1.234}$,

$$\rho_{1.234} = \sqrt{Ch_{max}(V_{1.234})} = \sqrt{0.612946} = 0.782908$$

and the associated characteristic vector is

$$[-0.534759 \quad 0.622277 \quad 0.571665]$$

With the same process we obtain, for $\rho_{2.134}$,

$$V_{2.134} = R^{*}_{12} \, R^{*-1}_{11} \, R^{*\prime}_{12}$$

$$= \begin{bmatrix} 0.233994 & 0.237671 & 0.040383 \\ 0.237671 & 0.428806 & 0.077936 \\ 0.040383 & 0.077936 & 0.164170 \end{bmatrix}$$

and $\rho_{2.134} = \sqrt{Ch_{max}(V_{2.134})} = \sqrt{0.605516} = 0.778149$

and the associated characteristic vector is

$$[0.543520 \quad 0.816636 \quad 0.194136]$$

For $\rho_{3.124}$,

$$V_{3.124} = R^{*}_{13} \, R^{*-1}_{11} \, R^{*\prime}_{13}$$

$$= \begin{bmatrix} 0.402631 & -0.132158 \\ -0.132158 & 0.151799 \end{bmatrix}$$

$$\rho_{3.124} = \sqrt{Ch_{max}(V_{3.124})} = \sqrt{0.459410} = 0.677798$$

and the assicuated characteristic vector is

$$[0.918793 \quad -0.394738]$$

For $\rho_{4.123}$,

$$V_{4.123} = R^{*}_{14} \, R^{*-1}_{11} \, R^{*\prime}_{14}$$

$$= \begin{bmatrix} 0.384348 & 0.092996 \\ 0.092996 & 0.144838 \end{bmatrix}$$

$$\rho_{4.123} = \sqrt{Ch_{max}(V_{4.123})} = \sqrt{0.416216} = 0.645148$$

and the associated characteristic vector is

$$[0.945997 \quad 0.324174]$$

Therefore the inverse of P is

$$P^{-1} = \begin{bmatrix} 2.58362 & -1.48289 & -0.48121 & -0.37908 \\ -1.48289 & 2.53496 & -0.42064 & -0.36871 \\ -0.48121 & -0.42064 & 1.84983 & -0.69221 \\ -0.37908 & -0.36871 & -0.69221 & 1.71296 \end{bmatrix}$$

and

$$(P^{-1})^{-1} = \begin{bmatrix} 1.23264 & 0.98365 & 0.85492 & 0.82999 \\ 0.98365 & 1.22961 & 0.84355 & 0.82324 \\ 0.85492 & 0.84355 & 1.28837 & 0.89140 \\ 0.82999 & 0.82324 & 0.89140 & 1.30488 \end{bmatrix}$$

Compare with (6.3.1) and (6.3.2), here we can see that $(P^{-1})^{-1}$ is not a correlation matrix. If it is normalized, it would be close to the minimum-determinant correlation matrix, though.

The reduction to the E matrix, we have from the Fletcher -Powell method, the set of canonical weights:

$$\underline{a}_1' = [-0.531 \quad 0.619 \quad 0.577]$$
$$\underline{a}_2' = [\ 0.561 \quad 0.804 \quad 0.192]$$
$$\underline{a}_3' = [\ 0.946 \quad -0.321]$$
$$\underline{a}_4' = [\ 0.958 \quad 0.285]$$

and the marginal totals for each set are

$$\underline{n}_1' = [\ 784 \quad 492 \quad 512 \quad 670]$$
$$\underline{n}_2' = [\ 727 \quad 789 \quad 485 \quad 447]$$
$$\underline{n}_3' = [\ 700 \quad 779 \quad 976]$$
$$\underline{n}_4' = [\ 716 \quad 985 \quad 757]$$

and the grand total: $n = 2458$

The categorical weights for the E matrix are

$$\underline{w}_1' = [-0.776065 \quad 1.527350 \quad 0.832238 \quad -0.849448]$$

$$\underline{w}_2' = [\ 0.866107 \quad 0.683918 \quad -0.946218 \quad -1.568864]$$

$$\underline{w}_3' = [\ 1.500610 \quad -1.023690 \quad -0.258397]$$

$$\underline{w}_4' = [\ 1.495060 \quad -0.317612 \quad -1.000810]$$

6.4 Four Categorical Variables, With 4, 4, 4 and 4 States

Let us consider a case of four sets with each set having four states.

The contingency table is

Table 6.4.1

		A_1				A_2				A_3				A_4			
		B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4
C_1	D_1	10	25	20	5	45	20	30	25	15	30	20	20	10	0	10	10
	D_2	5	10	5	5	20	10	25	30	30	20	10	5	5	10	10	5
	D_3	15	15	10	15	10	10	10	10	30	5	20	5	30	20	25	10
	D_4	10	15	15	15	30	35	30	30	20	30	30	30	10	5	0	20
C_2	D_1	40	40	0	30	5	10	40	35	10	5	0	5	0	0	10	20
	D_2	5	10	10	5	25	25	25	30	10	20	15	15	10	20	20	10
	D_3	25	35	35	40	10	10	5	5	10	10	10	10	20	10	20	20
	D_4	25	30	30	5	20	5	35	25	5	5	5	5	0	10	0	0
C_3	D_1	40	20	30	5	5	0	0	10	25	10	5	20	30	30	10	20
	D_2	40	10	10	5	35	10	20	10	20	20	5	5	20	15	15	15
	D_3	10	5	20	30	25	25	25	10	10	5	5	10	10	0	0	5
	D_4	25	30	20	10	10	10	10	20	5	10	5	10	0	10	20	10
C_4	D_1	0	20	10	0	20	25	0	0	40	35	40	40	5	10	10	20
	D_2	30	10	20	10	10	20	0	10	10	20	10	10	20	15	15	5
	D_3	25	30	10	10	20	20	5	5	20	10	5	0	30	25	30	30
	D_4	5	40	10	0	0	10	10	0	10	5	5	5	10	10	20	10

We have six two-way tables:

Table 6.4.2

A vs B

	A_1	A_2	A_3	A_4	Sub-total
B_1	310	290	270	210	1080
B_2	345	245	240	190	1020
B_3	255	270	190	215	930
B_4	190	255	195	220	860
Sub-total	1100	1060	895	835	3890

Table 6.4.3

A vs C

	A_1	A_2	A_3	A_4	Sub-total
C_1	195	370	320	190	1075
C_2	365	310	140	170	985
C_3	310	225	170	210	915
C_4	230	155	265	265	915
Sub-total	1100	1060	895	835	3890

Table 6.4.4

A vs D

	A_1	A_2	A_3	A_4	Sub-total
D_1	295	270	320	195	1080
D_2	190	305	225	210	930
D_3	330	205	165	285	985
D_4	285	280	185	145	895
Sub-total	1100	1060	895	835	3890

Table 6.4.5

B vs C

	B_1	B_2	B_3	B_4	Sub-total
C_1	295	260	270	250	1075
C_2	220	245	260	260	985
C_3	310	210	200	195	915
C_4	255	305	200	155	915
Sub-tOtal	1080	1020	930	860	3890

Table 6.4.6

B vs D

	B_1	B_2	B_3	B_4	Sub-total
D_1	300	280	235	265	1080
D_2	295	245	215	175	930
D_3	300	235	235	215	985
D_4	185	260	245	205	895
Sub-total	1080	1020	930	860	3890

Table 6.4.7

C vs D

	C_1	C_2	C_3	C_4	Sub-total
D_1	295	250	260	275	1080
D_2	215	255	245	215	930
D_3	240	275	195	275	985
D_4	325	205	215	150	895
Sub-total	1075	985	915	915	3890

The E_{ij} matrices are as follow:

$$E_{11} = \begin{bmatrix} 788.946 & -299.742 & -253.084 & -236.118 \\ -299.742 & 771.156 & -243.881 & -227.532 \\ -253.084 & -243.881 & 689.081 & -192.114 \\ -236.118 & -227.532 & -192.114 & 655.764 \end{bmatrix}$$

$$E_{22} = \begin{bmatrix} 780.154 & -283.187 & -258.200 & -238.766 \\ -283.187 & 752.545 & -243.856 & -225.501 \\ -258.200 & -243.856 & 707.660 & -205.604 \\ -238.766 & -225.501 & -205.604 & 669.871 \end{bmatrix}$$

$$E_{33} = \begin{bmatrix} 777.924 & -272.204 & -252.859 & -252.859 \\ -272.204 & 735.584 & -231.690 & -231.690 \\ -252.859 & -231.690 & 699.775 & -215.224 \\ -252.859 & -231.690 & -215.224 & 699.775 \end{bmatrix}$$

$$E_{44} = \begin{bmatrix} 780.154 & -258.200 & -273.470 & -248.483 \\ -258.200 & 707.660 & -235.488 & -213.973 \\ -273.470 & -235.488 & 735.584 & -226.625 \\ -248.483 & -213.973 & -226.625 & 689.081 \end{bmatrix}$$

$$E_{12} = \begin{bmatrix} 4.602 & 56.568 & -7.982 & -53.188 \\ -4.293 & -32.943 & 16.581 & 20.656 \\ 21.516 & 5.321 & -23.972 & -2.866 \\ -21.825 & -28.946 & 15.373 & 35.398 \end{bmatrix}$$

$$E_{13} = \begin{bmatrix} -108.984 & 86.465 & 51.259 & -28.740 \\ 77.069 & 41.594 & -24.332 & -94.332 \\ 72.667 & -86.626 & -40.521 & 54.479 \\ -40.752 & -41.433 & 13.593 & 68.593 \end{bmatrix}$$

$$E_{23} = \begin{bmatrix} -3.458 & -53.470 & 55.964 & 0.964 \\ -21.877 & -13.277 & -29.923 & 65.077 \\ 12.995 & 24.512 & -18.753 & -18.753 \\ 12.339 & 42.512 & -7.288 & -47.288 \end{bmatrix}$$

$$E_{14} = \begin{bmatrix} -10.398 & -72.982 & 51.465 & 31.915 \\ -24.293 & 51.581 & -63.406 & 36.118 \\ 71.517 & 11.028 & -61.626 & -20.919 \\ -36.825 & 10.373 & 73.567 & -47.114 \end{bmatrix}$$

$$E_{24} = \begin{bmatrix} 0.154 & 36.799 & 26.529 & -63.483 \\ -3.188 & 1.144 & -23.278 & 25.321 \\ -23.201 & -7.339 & -0.488 & 31.028 \\ 26.234 & -30.604 & -2.764 & 7.134 \end{bmatrix}$$

$$E_{34} = \begin{bmatrix} -3.458 & -42.005 & -32.204 & 77.667 \\ -23.470 & 19.512 & 25.585 & -21.626 \\ 5.964 & 26.247 & -36.690 & 4.479 \\ 20.964 & -3.753 & 43.309 & -60.521 \end{bmatrix}$$

The T conditional inverses are

$$T_1^{(-1)} = \begin{bmatrix} 0.035602 & 0.0 & 0.0 & 0.0 \\ 0.014819 & 0.039005 & 0.0 & 0.0 \\ 0.024891 & 0.024891 & 0.048113 & 0.0 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 0.035802 & 0.0 & 0.0 & 0.0 \\ 0.014240 & 0.039231 & 0.0 & 0.0 \\ 0.024579 & 0.024579 & 0.047308 & 0.0 \end{bmatrix}$$

$$T_3^{(-1)} = \begin{bmatrix} 0.035853 & 0.0 & 0.0 & 0.0 \\ 0.013858 & 0.039518 & 0.0 & 0.0 \\ 0.023376 & 0.023376 & 0.046752 & 0.0 \end{bmatrix}$$

$$T_4^{(-1)} = \begin{bmatrix} 0.039802 & 0.0 & 0.0 & 0.0 \\ 0.013268 & 0.040089 & 0.0 & 0.0 \\ 0.024195 & 0.024195 & 0.046179 & 0.0 \end{bmatrix}$$

Hence the super-matrix R consists of

$$R_{11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = R_{22} = R_{33} = R_{44}$$

$$R_{12} = \begin{bmatrix} 0.005865 & 0.081342 & 0.040084 \\ -0.003554 & -0.018937 & -0.011582 \\ 0.037339 & 0.047965 & 0.001943 \end{bmatrix}$$

$$R_{13} = \begin{bmatrix} -0.139114 & 0.067997 & 0.066579 \\ 0.049874 & 0.133985 & 0.091539 \\ 0.096872 & -0.001381 & -0.019569 \end{bmatrix}$$

$$R_{23} = \begin{bmatrix} -0.004438 & -0.077363 & 0.046031 \\ -0.032536 & -0.063223 & -0.068813 \\ -0.000284 & -0.019118 & -0.022983 \end{bmatrix}$$

$$R_{14} = \begin{bmatrix} -0.013254 & -0.109077 & 0.012789 \\ -0.039442 & 0.022682 & -0.083134 \\ 0.092277 & 0.034114 & -0.088339 \end{bmatrix}$$

$$R_{24} = \begin{bmatrix} 0.000198 & 0.052892 & 0.075873 \\ -0.004399 & 0.021178 & -0.013933 \\ -0.041965 & 0.007917 & -0.011572 \end{bmatrix}$$

$$R_{34} = \begin{bmatrix} -0.044383 & -0.062021 & -0.092759 \\ -0.034918 & -0.005315 & 0.007130 \\ -0.012554 & 0.023462 & -0.077876 \end{bmatrix}$$

Through some initialization of the canonical weights and the Fletcher-Powell method, a set of normalized canonical weights is obtained:

$$\underline{a}_1' = [\ 0.920 \quad 0.210 \quad -0.329]$$

$$\underline{a}_2' = [\ 0.219 \quad 0.854 \quad 0.470]$$

$$\underline{a}_3' = [-0.710 \quad 0.492 \quad 0.502]$$

$$\underline{a}_4' = [\ 0.215 \quad 0.975 \quad -0.044]$$

and P with elements $\underline{a}_i' R_{ij} \underline{a}_j$ equals

$$\begin{bmatrix} 1.00000 & 0.06358 & 0.19486 & -0.11623 \\ 0.06358 & 1.00000 & -0.04882 & 0.02758 \\ 0.19486 & -0.04882 & 1.00000 & 0.04619 \\ -0.11623 & 0.02758 & 0.04619 & 1.00000 \end{bmatrix} \quad (6.4.1)$$

By taking all elements positive, we obtain the inverse of P as

$$P^{-1} = \begin{bmatrix} 1.05500 & -0.05434 & -0.19776 & -0.11199 \\ -0.05434 & 1.00583 & -0.03760 & -0.01969 \\ -0.19776 & -0.03760 & 1.04146 & -0.02409 \\ -0.11199 & -0.01969 & -0.02409 & 1.01467 \end{bmatrix} \quad (6.4.2)$$

and $|P| = 0.939870$

as in the previous cases, we calculate the canonical-partial correlations.

For $\rho_{12.34}$:

$$R_{11}^* = \begin{bmatrix} 0.957258 & -0.005416 & 0.021044 \\ -0.005416 & 0.964167 & -0.006669 \\ 0.021044 & -0.006669 & 0.973430 \end{bmatrix}$$

$$F^*_{12} = \begin{bmatrix} 0.013196 & 0.088146 & 0.043467 \\ 0.007359 & -0.004897 & -0.013649 \\ 0.043187 & 0.047772 & 0.004041 \end{bmatrix}$$

$$R^*_{22} = \begin{bmatrix} 0.982784 & -0.001168 & 0.000141 \\ -0.001168 & 0.989249 & -0.003591 \\ 0.000141 & -0.003591 & 0.997012 \end{bmatrix}$$

$$T_1^{(-1)} = \begin{bmatrix} 1.022080 & 0.0 & 0.0 \\ 0.005762 & 1.018420 & 0.0 \\ -0.022248 & 0.006889 & 1.013820 \end{bmatrix}$$

$$U_{12.34} = \begin{bmatrix} 0.010402 & 0.000331 & 0.004984 \\ 0.000331 & 0.000279 & 0.000173 \\ 0.004984 & 0.000173 & 0.004125 \end{bmatrix}$$

$$\rho_{12.34} = \sqrt{Ch_{max}(U_{12.34})} = \sqrt{0.013164} = 0.114734$$

For $\rho_{13.24}$,

$$R^*_{11} = \begin{bmatrix} 0.978976 & 0.004380 & 0.001544 \\ 0.004380 & 0.990508 & -0.003621 \\ 0.001544 & -0.003621 & 0.978542 \end{bmatrix}$$

$$R^*_{13} = \begin{bmatrix} -0.142043 & 0.073826 & 0.076331 \\ 0.042719 & 0.132311 & 0.082712 \\ 0.092468 & 0.008985 & -0.024865 \end{bmatrix}$$

$$R^*_{33} = \begin{bmatrix} 0.986434 & -0.001441 & -0.008445 \\ -0.001441 & 0.988213 & -0.000748 \\ -0.008445 & -0.000748 & 0.985059 \end{bmatrix}$$

$$T_1^{(-1)} = \begin{bmatrix} 1.010680 & 0.0 & 0.0 \\ -0.004496 & 1.004790 & 0.0 \\ -0.001611 & 0.003702 & 1.010910 \end{bmatrix}$$

$$U_{13.24} = \begin{bmatrix} 0.032355 & 0.010070 & -0.014801 \\ 0.010070 & 0.026772 & 0.003393 \\ -0.014801 & 0.003393 & 0.009617 \end{bmatrix}$$

$$\rho_{13.24} = \sqrt{Ch_{max}(U_{13.24})} = \sqrt{0.044337} = 0.208264$$

For $\rho_{23.14}$:

$$R_{22}^{*} = \begin{bmatrix} 0.989519 & -0.002900 & -0.000169 \\ -0.002900 & 0.989652 & -0.003969 \\ -0.000169 & -0.003969 & 0.996284 \end{bmatrix}$$

$$R_{23}^{*} = \begin{bmatrix} 0.003416 & -0.078666 & 0.050938 \\ -0.023632 & -0.066147 & -0.073388 \\ 0.004099 & -0.024379 & -0.028086 \end{bmatrix}$$

$$R_{33}^{*} = \begin{bmatrix} 0.955708 & 0.003819 & 0.003177 \\ 0.003819 & 0.976286 & -0.015679 \\ 0.003177 & -0.015679 & 0.980485 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 1.005280 & 0.0 & 0.0 \\ 0.002946 & 1.005210 & 0.0 \\ 0.000183 & 0.004019 & 1.001870 \end{bmatrix}$$

$$U_{23.14} = \begin{bmatrix} 0.008963 & 0.001513 & 0.000549 \\ 0.001513 & 0.010817 & 0.003782 \\ 0.000549 & 0.003782 & 0.001491 \end{bmatrix}$$

$$\rho_{23.14} = \sqrt{Ch_{max}(U_{23.14})} = \sqrt{0.012828} = 0.113259$$

For $\rho_{14.23}$,

$$R_{11}^{*} = \begin{bmatrix} 0.962255 & -0.008770 & 0.010206 \\ -0.008770 & 0.970902 & -0.002864 \\ 0.010206 & -0.002864 & 0.986228 \end{bmatrix}$$

$$R_{14}^{*} = \begin{bmatrix} -0.008134 & -0.121429 & 0.006441 \\ -0.032708 & 0.024110 & -0.072400 \\ 0.092943 & 0.037540 & -0.082849 \end{bmatrix}$$

$$R_{44}^{*} = \begin{bmatrix} 0.996695 & 0.000454 & -0.001722 \\ 0.000454 & 0.992442 & -0.007429 \\ -0.001722 & -0.007429 & 0.978240 \end{bmatrix}$$

$$T_1^{(-1)} = \begin{bmatrix} 1.019420 & 0.0 & 0.0 \\ 0.009250 & 1.014910 & 0.0 \\ -0.010654 & 0.002874 & 1.007010 \end{bmatrix}$$

$$U_{14.23} = \begin{bmatrix} 0.015541 & -0.003059 & -0.006137 \\ -0.003059 & 0.007154 & 0.004036 \\ -0.006137 & 0.004036 & 0.017416 \end{bmatrix}$$

$$\rho_{14.23} = \sqrt{Ch_{max}(U_{14.23})} = \sqrt{0.024187} = 0.155521$$

For $\rho_{24.13}$,

$$R_{22}^{*} = \begin{bmatrix} 0.990316 & -0.004398 & -0.000835 \\ -0.004398 & 0.980284 & -0.006593 \\ -0.000835 & -0.006593 & 0.996928 \end{bmatrix}$$

$$R_{24}^{*} = \begin{bmatrix} -0.005357 & 0.050317 & 0.082929 \\ -0.011255 & 0.028891 & -0.018434 \\ -0.041961 & 0.013146 & -0.012144 \end{bmatrix}$$

$$R^*_{44} = \begin{bmatrix} 0.988692 & -0.004633 & 0.003997 \\ -0.004633 & 0.978641 & 0.001890 \\ 0.003997 & 0.001890 & 0.973226 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 1.004870 & 0.0 & 0.0 \\ 0.004486 & 1.010010 & 0.0 \\ 0.000874 & 0.006740 & 1.001560 \end{bmatrix}$$

$$U_{24.13} = \begin{bmatrix} 0.009762 & 0.000016 & -0.000121 \\ 0.000016 & 0.001354 & 0.001108 \\ -0.000121 & 0.001108 & 0.002121 \end{bmatrix}$$

$$\rho_{24.13} = \sqrt{Ch_{max}(U_{24.13})} = \sqrt{0.0097635} = 0.098811$$

For $\rho_{34.12}$,

$$R^*_{33} = \begin{bmatrix} 0.968078 & 0.000799 & 0.005244 \\ 0.000799 & 0.966448 & -0.018627 \\ 0.005244 & -0.018627 & 0.978744 \end{bmatrix}$$

$$R^*_{34} = \begin{bmatrix} -0.013308 & -0.080585 & -0.077975 \\ -0.030405 & 0.005383 & 0.022301 \\ -0.007823 & 0.029282 & -0.077701 \end{bmatrix}$$

$$R^*_{44} = \begin{bmatrix} 0.987970 & -0.002779 & 0.004747 \\ -0.002779 & 0.982688 & 0.002651 \\ 0.004747 & 0.002651 & 0.978633 \end{bmatrix}$$

$$T_3^{(-1)} = \begin{bmatrix} 1.016350 & 0.0 & 0.0 \\ -0.000841 & 1.017210 & 0.0 \\ -0.005492 & 0.019490 & 1.001100 \end{bmatrix}$$

$$U_{34.12} = \begin{bmatrix} 0.013389 & -0.001878 & 0.003876 \\ -0.001878 & 0.001533 & -0.001387 \\ 0.003876 & -0.001387 & 0.007170 \end{bmatrix}$$

$$\rho_{34.12} = \sqrt{Ch_{max}(U_{34.12})} = \sqrt{0.015622} = 0.124989$$

Hence the canonical-partial correlation matrix is

$$\begin{bmatrix} 1.00000 & 0.11473 & 0.20826 & 0.15552 \\ 0.11473 & 1.00000 & 0.11326 & 0.09881 \\ 0.20826 & 0.11326 & 1.00000 & 0.12499 \\ 0.15552 & 0.09881 & 0.12499 & 1.00000 \end{bmatrix}$$

We will find all of the canonical-multiple correlation of each set vs the others:

For $\rho_{1.234}$:

$$V_{1.234} = R^{*}_{12} R^{*-1}_{22} R^{*\prime}_{12}$$

$$= \begin{bmatrix} 0.052699 & 0.005677 & -0.016017 \\ 0.005678 & 0.036098 & 0.006816 \\ -0.016017 & 0.006816 & 0.0[illegible]279 \end{bmatrix}$$

$$\rho_{1.234} = \sqrt{Ch_{max}(V_{1.234})} = \sqrt{0.06[illegible]196} = 0.24758[illegible]$$

and the associated characteristic vector is

$$[\, 0.889727 \quad 0.0786[illegible] \quad -0.449626 \,]$$

For $\rho_{2.134}$:

$$V_{2.134} = R^{*}_{12} R^{*-1}_{11} R^{*\prime}_{12}$$

$$= \begin{bmatrix} 0.019350 & 0.004[illegible] & 0.000706 \\ 0.004371 & 0.0210[illegible] & 0.007679 \\ 0.000706 & 0.0076[illegible] & 0.005171 \end{bmatrix}$$

$$\rho_{2.134} = \sqrt{Ch_{max}(V_{2.134})} = \sqrt{0.026699} = 0.163401$$

For $\rho_{3.124}$:

$$V_{3.124} = R^{*}_{13} \, R^{*-1}_{11} \, R^{*\prime}_{13}$$

$$= \begin{bmatrix} 0.044884 & -0.002606 & -0.001367 \\ -0.002606 & 0.035029 & 0.017244 \\ -0.001367 & 0.017244 & 0.028365 \end{bmatrix}$$

$$\rho_{3.124} = \sqrt{Ch_{max}(V_{3.124})} = \sqrt{0.050691} = 0.225145$$

and the associated characteristic vector is

$$[\ -0.444899 \quad 0.695402 \quad 0.564339\ ]$$

For $\rho_{4.123}$:

$$V_{4.123} = R^{*}_{14} \, R^{*-1}_{11} \, R^{*\prime}_{14}$$

$$= \begin{bmatrix} 0.013239 & 0.003462 & -0.003724 \\ 0.003462 & 0.024959 & 0.001623 \\ -0.003724 & 0.001623 & 0.034200 \end{bmatrix}$$

$$\rho_{4.123} = \sqrt{Ch_{max}(V_{4.123})} = \sqrt{0.034951} = 0.186952$$

and the associated characteristic vector is

$$[\ 0.151437 \quad 0.170171 \quad 0.982639\ ]$$

Therefore the inverse of P is

$$P^{-1} = \begin{bmatrix} 1.06529 & 0.12003 & 0.22062 & 0.16340 \\ 0.12003 & 1.02743 & 0.11783 & 0.10195 \\ 0.22062 & 0.11783 & 1.05339 & 0.13059 \\ 0.16340 & 0.10195 & 0.13059 & 1.03621 \end{bmatrix}$$

and

$$(P^{-1})^{-1} = \begin{bmatrix} 1.00604 & -0.08365 & -0.18560 & -0.12702 \\ -0.08365 & 1.00022 & -0.08513 & -0.07449 \\ -0.18560 & -0.08513 & 1.00879 & -0.08949 \\ -0.12702 & -0.07449 & -0.08949 & 1.00368 \end{bmatrix}$$

and $|P| = 0.93001$

compare with (6.4.1) and (6.4.2) again the approximation is quite close.

For the reduction to the categorical weights for E matrix, from Fletcher-Powell method we obtain a set of normalized canonical weights:

$$\underline{a}_1' = [\ 0.920 \quad 0.210 \quad -0.329]$$
$$\underline{a}_2' = [\ 0.219 \quad 0.854 \quad 0.470]$$
$$\underline{a}_3' = [-0.710 \quad 0.492 \quad 0.502]$$
$$\underline{a}_4' = [\ 0.215 \quad 0.975 \quad -0.044]$$

and the marginal totals for each set are:

$$\underline{n}_1' = [1100 \quad 1060 \quad 895 \quad 835]$$
$$\underline{n}_2' = [1080 \quad 1020 \quad 930 \quad 860]$$
$$\underline{n}_3' = [1075 \quad 985 \quad 915 \quad 915]$$
$$\underline{n}_4' = [1080 \quad 930 \quad 985 \quad 895]$$

the grand total $n = 3890$.

The sets of categorical weights for the entire E matrix are therefore

$$\underline{w}_1' = [\ 1.465660 \quad -0.259803 \quad -1.250040 \quad -0.261131]$$
$$\underline{w}_2' = [\ 0.353890 \quad 1.196350 \quad -0.228598 \quad -1.616140]$$
$$\underline{w}_3' = [-1.149300 \quad 1.228670 \quad 0.746598 \quad -0.718985]$$
$$\underline{w}_4' = [\ 0.347959 \quad 1.498080 \quad -1.001110 \quad -0.874771]$$

6.5 A Validation Study

Five hundred observations were generated, random normal numbers from a tri-variate normal distribution. Uncorrelated random normal numbers were multiplied by the triangular matrix

$$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.8 & 0.6 & 0.0 \\ 0.0 & 0.6 & 0.8 \end{bmatrix}$$

thus, the sample is from a normal distribution with mean vector $\underline{0}$ and variance-covariance matrix

$$\Sigma = \begin{bmatrix} 1.00 & 0.80 & 0.00 \\ 0.80 & 1.00 & 0.36 \\ 0.00 & 0.36 & 1.00 \end{bmatrix}.$$

Each of the three variables was divided into four slices

$$Y_1 : \quad < -0.7 \ , \ -0.7 \text{ to } +0.3 \ , \ +0.3 \text{ to } +1 \ , \quad > +1 \ .$$

The first category was given the value 3, the second was given the value 4, the third was given the value 1 and the last was given the value 2. Thus, the expected value under the slice "3" (< -0.7) was $\frac{-z_1}{f_1}$ where z_1 is the ordinate under a stand normal at $x = -0.7$ and f_1 is the cumulative distribution function of the standard normal at $x = -0.7$. For the next slice "4", the expected value is $\frac{z_1 - z_2}{f_2 - f_1}$ where z_2 and f_2 are ordinate and area for $x = +0.3$ (the second partition). Similar partitions were made for the other two random variables. The results are

Table 6.5.1

Categories and Expected Values

Y_1 (levels)	3	4	1	2
Slice	< -0.7	-0.7 to 0.3	0.3 to 1	> 1
Expected	-1.290	-0.184	0.624	1.525

Y_2 (levels)	1	4	3	2
Slice	< -1.1	-1.1 to 0	0 to 0.5	> 0.5
Expected	-1.606	-0.497	0.244	1.141

Y_3 (levels)	2	3	1	4
Slice	< -1.2	-1.2 to 0.3	0.3 to 1.2	> 1.2
Expected	-1.687	-0.372	0.701	1.687

Contingency tables were obtained and entered into the program for categorical scaling: A_1, A_2, A_3, A_4 corresponds to levels 1, 2, 3, 4 of Y_1; B corresponds to Y_2 and C corresponds to Y_3.

Table 6.5.2

	A_1				A_2				A_3				A_4			
	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4
C_1	0	28	3	3	0	18	0	0	7	1	0	22	2	6	14	11
C_2	1	2	5	10	1	3	1	2	13	0	0	2	5	0	1	18
C_3	1	18	19	17	0	30	9	3	37	0	0	29	9	2	30	57
C_4	0	14	1	0	0	10	0	0	3	0	2	7	0	10	8	5

We have three two-way tables:

Table 6.5.3

A vs B

	A_1	A_2	A_3	A_4	Sub-total
B_1	2	1	60	16	79
B_2	62	61	1	18	142
B_3	28	10	2	53	93
B_4	30	5	60	91	186
Sub-total	122	77	123	178	500

Table 6.5.4

A vs C

	A_1	A_2	A_3	A_4	Sub-total
C_1	34	18	30	33	115
C_2	18	7	15	24	64
C_3	55	42	66	98	261
C_4	15	10	12	23	60
Sub-total	122	77	123	178	500

Table 6.5.5

B vs C

	B_1	B_2	B_3	B_4	Sub-total
C_1	9	43	17	36	115
C_2	20	5	7	32	64
C_3	47	50	58	106	261
C_4	3	34	11	12	60
Sub-total	79	132	93	186	500

The E matrices are as follows:

$$E_{11} = \begin{bmatrix} 92.232 & -18.788 & -30.012 & -43.432 \\ -18.788 & 65.142 & -18.942 & -27.412 \\ -30.012 & -18.942 & 92.742 & -43.788 \\ -43.432 & -27.412 & -43.788 & 114.632 \end{bmatrix}$$

$$E_{22} = \begin{bmatrix} 66.518 & -22.436 & -14.694 & -29.388 \\ -22.436 & 101.672 & -26.412 & -52.824 \\ -14.694 & -26.412 & 75.702 & -34.596 \\ -29.388 & -52.824 & -34.596 & 116.808 \end{bmatrix}$$

$$E_{33} = \begin{bmatrix} 88.550 & -14.720 & -60.030 & -13.800 \\ -14.720 & 55.808 & -33.408 & -7.680 \\ -60.030 & -33.408 & 124.758 & -31.320 \\ -13.800 & -7.680 & -31.320 & 52.800 \end{bmatrix}$$

$$E_{12} = \begin{bmatrix} -17.276 & 27.352 & 5.308 & -15.384 \\ -11.166 & 39.132 & -4.322 & -23.644 \\ 40.566 & -33.932 & -20.878 & 14.244 \\ -12.124 & -32.552 & 19.892 & 24.784 \end{bmatrix}$$

$$E_{13} = \begin{bmatrix} 5.940 & 2.384 & -8.684 & 0.360 \\ 0.290 & -2.856 & 1.806 & 0.760 \\ 1.710 & -0.744 & 1.794 & -2.760 \\ -7.940 & 1.216 & 5.034 & 1.640 \end{bmatrix}$$

$$E_{23} = \begin{bmatrix} -9.170 & 9.888 & 5.762 & -6.480 \\ 20.340 & -13.176 & -24.124 & 16.960 \\ -4.390 & -4.904 & 9.454 & -0.160 \\ -6.780 & 8.192 & 8.908 & -10.320 \end{bmatrix}$$

The T conditional inverses are:

$$T_1^{(-1)} = \begin{bmatrix} 0.104126 & 0.0 & 0.0 & 0.0 \\ 0.026015 & 0.127707 & 0.0 & 0.0 \\ 0.047914 & 0.047914 & 0.117252 & 0.0 \end{bmatrix}$$

$$T_2^{(-1)} = \begin{bmatrix} 0.122611 & 0.0 & 0.0 & 0.0 \\ 0.034768 & 0.103084 & 0.0 & 0.0 \\ 0.042333 & 0.042333 & 0.127000 & 0.0 \end{bmatrix}$$

$$T_3^{(-1)} = \begin{bmatrix} 0.106268 & 0.0 & 0.0 & 0.0 \\ 0.022757 & 0.136895 & 0.0 & 0.0 \\ 0.116410 & 0.116410 & 0.143171 & 0.0 \end{bmatrix}$$

Hence the super-matrix R consists of

$$R_{11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = R_{22} = R_{33}$$

$$R_{12} = \begin{bmatrix} -0.220563 & 0.231044 & 0.114608 \\ -0.229946 & 0.523304 & 0.109727 \\ 0.416104 & 0.036240 & -0.194803 \end{bmatrix}$$

$$R_{13} = \begin{bmatrix} 0.065728 & 0.048057 & -0.028562 \\ 0.020357 & -0.037081 & -0.12262 \\ 0.053028 & -0.003682 & 0.028235 \end{bmatrix}$$

$$R_{23} = \begin{bmatrix} -0.119482 & 0.140382 & 0.111396 \\ 0.188935 & -0.098413 & -0.238482 \\ -0.008997 & -0.106240 & -0.037952 \end{bmatrix}$$

In order to initialize the iterative minimum-determinant solution, we find the canonical-multiple correlations of each set vs the others combined.

For $\rho_{1.23}$,

$$V_{1.23} = R^*_{12} R^{*-1}_{22} R^{*\prime}_{12}$$

$$= \begin{bmatrix} 0.132787 & 0.201982 & -0.111764 \\ 0.201982 & 0.379307 & -0.118215 \\ -0.117641 & -0.118215 & 0.229624 \end{bmatrix}$$

$$\rho_{1.23} = \sqrt{Ch_{max}(V_{1.23})} = \sqrt{0.569450} = 0.754619$$

and the associated characteristic vector is

$$[\ 0.471882 \quad 0.769173 \quad -0.430927]$$

For $\rho_{2.13}$,

$$V_{2.13} = R^*_{12} R^{*-1}_{11} R^{*\prime}_{12}$$

$$= \begin{bmatrix} 0.318682 & -0.209537 & -0.148678 \\ -0.209537 & 0.414171 & 0.091512 \\ -0.148678 & 0.091512 & 0.075762 \end{bmatrix}$$

$$\rho_{2.13} = \sqrt{Ch_{max}(V_{2.13})} = \sqrt{0.630610} = 0.794109$$

and the associated characteristic vector is

$$[-0.624674 \quad 0.726168 \quad 0.287158]$$

For $\rho_{3.12}$,

$$V_{3.12} = R^*_{13} R^{*-1}_{11} R^{*\prime}_{13}$$

$$= \begin{bmatrix} 0.085558 & -0.065363 & -0.092459 \\ -0.065363 & 0.081871 & 0.078239 \\ -0.092459 & 0.078239 & 0.115964 \end{bmatrix}$$

$$\rho_{3.12} = \sqrt{Ch_{max}(V_{3.12})} = \sqrt{0.254623} = 0.504601$$

and the associated characteristic vector is

$$[\ 0.556117\ \ -0.508234\ \ -0.657594]$$

After 14 iterations by the Fletcher-Powell method we obtained the Minimum-Determinant Solution:

$$\underline{a}_1' = [\ 0.470 \quad 0.776 \quad -0.419]$$

$$\underline{a}_2' = [-0.624 \quad 0.726 \quad 0.286]$$

$$\underline{a}_3' = [\ 0.545 \quad -0.526 \quad -0.651]$$

The marginal totals for each set are:

$$\underline{n}_1' = [\ 122 \quad 77 \quad 123 \quad 178]$$

$$\underline{n}_2' = [\ 79 \quad 142 \quad 93 \quad 186]$$

$$\underline{n}_3' = [\ 115 \quad 64 \quad 261 \quad 60]$$

and the grand total is $n = 500$.

Re-translating these into original weights we obtain

$$\underline{w}_1' = [\ 0.828 \quad 1.498 \quad -1.370 \quad -0.269]$$

$$\text{(Expected value)} = [\ 0.624 \quad 1.525 \quad -1.290 \quad -0.184]$$

$$\underline{w}_2' = [-1.441 \quad 1.380 \quad 0.249 \quad -0.566]$$

$$\text{(Expected value)} = [-1.606 \quad 1.141 \quad 0.244 \quad -0.497]$$

$$\underline{w}_3' = [\ 0.998 \quad -1.643 \quad -0.421 \quad 1.667]$$

$$\text{(Expected value)} = [\ 0.[illegible]01 \quad -1.687 \quad -0.37[illegible] \quad 1.687]$$

It is thus seen that the theoretical values were quite adequately reproduced from the categorized data only, even though the latter were unordered.

The correlation matrix obtained from the scaled scores, and the contingency tables 6.5.3, 6.5.4 and 6.5.5 is

$$\begin{bmatrix} 1.000 & 0.712 & 0.038 \\ 0.712 & 1.000 & 0.380 \\ 0.038 & 0.380 & 1.000 \end{bmatrix}$$

The ρ_{13} and ρ_{23} values are quite adequate approximations to the true covariances.

$$P\left| r > 0.038 \mid \rho = 0 \right| = 0.603$$

$$P\left| r > 0.380 \mid \rho = 0.36 \right| = 0.695$$

(from exact distribution of r^2)

However, the 0.712 value is too low

$$P\left| r < 0.712 \mid \rho = 0.8 \right| < 10^{-4}$$

If the vectors $\underline{w}_1'$ and $\underline{w}_2'$ are replaced by the vectors of expected values, the resulting correlation between the A-set and the B-set is 0.7036, even less than the 0.712 obtained from the minimum-determinant solution. Hence the significant reduction of the high correlation is due to replacement of the continuous variables by four points, and not due to the method of analysis.

6.6 Comparison of Initial Trials

1. For the case (3 x 3 x 3)

4.2(2)	4.2(4)	4.2(5)	4.3
-0.23586	0.05636	0.01026	0.0547
0.97179	0.99841	0.99995	0.998
0.90665	0.91953	0.92686	0.929
-0.42188	-0.39303	-0.37540	-0.369
0.89415	0.87451	0.87617	0.856
0.44770	0.48501	0.48199	0.516

2. **For the case (2 x 5 x 2)**

4.2(2)	4.2(4)	4.2(5)	4.3 [1)]
1.00000	1.00000	1.00000	1.000
0.36902	0.44386	0.50172	0.501
0.62149	0.61376	0.59852	0.598
0.47613	0.49616	0.51773	0.517
0.47981	0.42439	0.34919	0.349
1.00000	1.00000	1.00000	1.000

3. For the case (4 x 4 x 3 x 3)

4.2(2)	4.2(4)	4.2(5)	4.3
-0.51993	-0.53287	-0.53476	-0.531
0.62955	0.62204	0.62220	0.619
0.57807	0.57367	0.57166	0.577
0.56258	0.55387	0.54352	0.561
0.80269	0.80935	0.81664	0.804
0.19798	0.19539	0.19414	0.192
0.93833	0.93138	0.91879	0.946
-0.34573	-0.36404	-0.39474	-0.321
0.96108	0.94873	0.94599	0.958
0.27627	0.31609	0.32417	0.284

---

1) 4.2(2) - Average canonical scales.
   4.2(4) - Multiple regression weights.
   4.2(5) - Canonical-multiple correlations as weights.
   4.3 - Minimum-determinant solution.

4. For the case (4 x 4 x 4 x 4)

4.2(2)	4.2(4)	4.2(5)	4.3
0.68971	0.70033	0.88973	0.920
0.43353	0.60842	0.07887	0.207
0.57996	0.37332	-0.44963	-0.331
0.68208	0.65639	0.50835	0.222
0.70489	0.72649	0.80568	0.852
0.19469	0.20338	0.30408	0.472
0.24854	-0.02083	-0.44489	-0.711
0.69108	0.71414	0.69540	0.491
0.67870	0.69968	0.56434	0.501
-0.12662	-0.18807	-0.15144	0.211
0.11516	-0.02826	0.10717	0.976
0.98524	0.98175	0.98264	-0.046

5. For the case (4 x 4 x 4)

4.2(2)	4.2(4)	4.2(5)	4.3
0.85941	0.24705	0.47188	0.470
0.51110	0.78577	0.76917	0.776
0.01350	-0.56704	-0.43093	-0.419
-0.59666	-0.61489	-0.62467	-0.624
0.76332	0.74052	0.72616	0.726
0.24764	0.27117	0.28716	0.286
0.30079	-0.80374	0.55612	0.545
0.81332	0.16229	-0.50823	-0.526
0.49802	0.57241	-0.65759	-0.651

For comparison between various forms of initial solutions with the best (canonical-mulitple correlations) initial solutions, the (4 x 4 x 4 x 4) study has been chosen:

With permissible error: EPS = 10^{-3}

(1) Initial weights are all <u>0.5</u>

<u>90</u> iterations to obtain convergence and the normalized canonical weights are

$$\begin{matrix} [\ 0.920 & 0.212 & -0.328] \\ [\ 0.221 & 0.855 & 0.469] \\ [-0.709 & 0.494 & 0.502] \\ [\ 0.216 & 0.975 & -0.040] \end{matrix}$$

(2) Initial weights are all <u>1.0</u>

<u>109</u> iterations to obtain convergence and the normalized canonical weights are

$$\begin{matrix} [\ 0.917 & 0.222 & -0.329] \\ [\ 0.245 & 0.848 & 0.468] \\ [-0.709 & 0.498 & 0.499] \\ [\ 0.196 & 0.980 & -0.024] \end{matrix}$$

(3) Initial weights are all <u>2.0</u>

<u>133</u> iterations to obtain convergence and the normalized canonical weights are

$$\begin{matrix} [\ 0.896 & 0.417 & -0.150] \\ [\ 0.796 & 0.470 & 0.382] \\ [-0.598 & 0.676 & 0.429] \\ [\ 0.060 & 0.927 & 0.363] \end{matrix}$$

Note: If EPS = 10^{-4} and the initial weights are either all 1.0 or the multiple-regression weights, after 150 iterations, the limit set by the computer program, no convergence can be obtained.

It is noted that, in every instance, the canonical-multiple correlation weights (each set against the totality of the others) gives a very good approximation to the final minimum-determinant solution.

When the relations between the categorical variables are strong, the multiple regression approch, based upon pair-wise canonical correlations is also quite useful (section 6.6). However, since it is easier to obtain canonical-multiple correlations there is really no reason for using this kind of initial solution.

The unweighted average of canonical correlations are useless as initial solution.

In conclusion it can be said that the canonical-multiple correlation weights appear to be so good that iterations to the minimum-determinant solution would be required only if unusually high accuracy is desired; this would be the exception in categorical data analysis.

CHAPTER VII

SUMMARY AND CONCLUSION

A generalization of the Fisher-Lancaster technique for scaling pairs of categorical variables has been studied in considerable detail. For the reason stated in CHAPTER III, R.G.D. Steel's criterion of a minimum-determinant solution was adopted as an optimal criterion. The solutions satisfying this criterion have been compared with various approximations, including:

(1) Averages and weighted averages of canonical vectors obtained by pairing each set with each other set;

(2) canonical vectors obtained by pairing each set with the totality of the other sets;

(3) canonical-partial and canonical-multiple correlations being used to construct the inverse of a correlation matrix of the derived scales.

Of the three methods, the second is the least time-consuming, and also consistently the best. It is so good, in fact, that for all practical purposes iteration to a minimum-determinant solution is unnecessary.

It is thus proposed that categorical scaling of k response variables be conducted as follows:

(1) From the k-dimensional contingency table construct a super-matrix D_y as described in section 4.1.

(Computer program C-E-R in Appendix A2).

(2) Obtain conditional inverses of pseudo-triangular matrices T_i ($T_i T_i' = E_{ii}$) as explained in section 4.1.

(3) Obtain a super-correlation matrix whose submatrices are $R_{ij} = T_i^{(-1)} E_{ij} T_j^{(-1)'}$ and $R_{ii} = I$. (Also performed in the computer program C-E-R).

(4) By pairing each set with the totality of the other sets, obtain "canonical-multiple" correlation and the characteristic vector associated with each. These are, for all practical purposes, the canonical weights from which the categorical scales can be obtained, as described in section 4.2.1.5 (For detailed description see illustration and the computer program CPCM in Appendix D).

(4b) If high precision is desired, use an iterative method such as the Fletcher-Powell method to obtain canonical weights that satisfy the minimum-determinant solution (see description in section 4.3 and the computer program FPM in Appendix C).

(5) By inverting the process in (3), and using the canonical weights (4) and (4b), obtain scales for each categorical variable. Standardization in such a way that, in the original set, the scaled responses would have mean zero and variance one is desirable

(see section 4.4 and the computer program RTE in Appendix E).

It is not claimed that the foregoing studies set to rest, once for all, the problem of multi-dimensional categorized scaling. Other criteria generalizing canonical correlations will probably produce somewhat different results. However, the close proximity of an ad-hoc result (canonical weights of each set vs. the totality of others) and a maximum-likelihood result (minimum-determinant solutions) seems to indicate that, whatever future improvement will be found, it will produce only slight changes of the final scales.

On the other hand, simple averaging techniques, which have been repeatedly used in the literature, often produce quite different results. If some author can find some method for determinimg weights by easier means than characteristic roots and vectors of relatively small matrices (4 by 4 for 5 states), and if these weights approximate ours in many different situations, his method should surely replace the five steps recommemded in this summary.

BIBLIOGRAPHY

[1] T.W. Anderson : An Introduction to Multivariate Statistical Analysis. John Wiley and Sons, N.Y. 1858.

[2] ____________ : Statistical Inference in Factor Analysis. Proceeding of the Third Symposium on Mathematical Statistics and Probability, 1965, 5, 11-50.

[3] F.M. Andrew and R.C. Messenger : Multivariate Nominal Scale Analysis. ISR, University of Michigan, Ann Arbor, Michigan, 1973.

[4] R.E. Bargmann : Interpretation and Use of a Generalized Discriminant Function. Essay in Probability and Statistics, Chapter 3, University of North Carolina Press, Chapel-Hill, N.C. 1969.

[5] ____________ : Matrices and Determinants. CRC Standard Mathematical Tables, 1966 or later,

[6] H. Cramér : Mathematical Method of Statistics. Princeton University Press, N.J. 1946.

[7] H. Edgerton and E.L. Kolbe : The Method of Minimal Variation for the Combination of Criteria. Psychometrika, 1963, 1, 183-187.

[8] P.H. Dubois : Multivariate Correlation Analysis. Harper, N.Y. 1957.

[9] R. Fletcher and M.J.D. Powell : A rapidly Convergent Descent Method for Minimization. The Computer Journal, 1963, 6, 163-168.

[10] R.A. Fisher : The Precision of Discriminant Function. Annal Eugenics, 1940, 10, 422-429.

[11] L.A. Goodman and W.H. Kruskal : Measures of Association of Cross Classification. Journal of American Statistical Association, 1954, 49, 732-764.

[12] J. Greenstadt : The Determination of the Characteristic Roots of A Matrix by the Jacobi Method. Mathematical Method for Digital Computer, Vol. I, John Wiley and Sons, N.Y.

[13] L.Hays : Statistics for Psychologist. Holt, Rinehart and Winston, N.Y. 1963.

[14] D.L. Hecks : Charts of Some Upper Percentage Points of the Largest Characteristic Root. Annals of Mathematical Statistics, 1960, 31, 625-642.

[15] P. Horst : Generalized Canonical Correlation. Journal of Clinical Psychology Monog. Supplement, 1964, 14.

[16] _______ : Obtain a Composite Measure from A Number of Different Measures of the Same Attribute. Psychometrika, 1963, 1, 53-60.

[17] H. Hotelling : The Most Predicable Criterion. Journal of Educational Psychology, 1935, 26, 139-142.

[18] __________ : Relation Between Two sets of Variates. Biometrika, 1936, 28, 321-377.

[19] W.G. Howe : Some Contribution to Factor Analysis, Ph.D. Dissertation, University of North Carolina, Chapel-Hill, N.C. 1956.

[20] E. Jenning : Matrix Formulas for Part and Partial Correlation. Psychometrika, 1965, 30, 353-356.

[21] G.W. Joe : Selected Topics in the Analysis of M-Sets of Variates. Ed.D. Dissertation, University of Georgia, Athens, Ga. 1969.

[22] M.G. Kendall and A. Stuart : The Advanced Theory of Statistics, Vol. II, Griffin and Co. London, 1948.

[23] K.R. Kundert and R.E. Bargmann : Tools of Analysis for Pattern Recognition. Technical Report, Department of Statistics and Computer Science, University of Georgia, Athens, Ga. 1972, 69.

[24] H.O. Lancaster : Some Properties of the Bivariate Normal Distribution Considered in the Form of A Contingency Table. Biometrika, 1957, 44, 289-291.

[25] ____________ : The Structure of Bivariate Distribution. Annals of Mathematical Statistics, 1958, 29, 719-736.

[26] ____________ : On tests of Independence in Several Dimensions. Journal of Australia Mathematical Society, 1960.

[27] J. Loevinger : A Systematic Approach to the Construction and Evaluation of Tests of Ability. Psychometrika, Monog. 1947, 61.

[28] F.M. Lord : Some Relation Between Guttman's Principal Component of Scale Analysis and Other Psychometric Theory. Psychometrika, 1958, 23, 291-195.

[29] J. McKeon : Canonical Analysis - Some Relation Between Canonical Correlation, Factor Analysis, Discriminant Function Analysis and Scaling Theory. Princeton ETS Monog. 18, (Undated).

[30] R.L. Messenger and L.M. Mandell : A Model Search Technique for Predictive Nominal Scale Multivariate Analysis. Journal of American Statistical Association, 1972, 67, 768-772.

[31] D.M. Morrison : Multivariate Statistical Method. McGraw-Hill Book Co. N.Y. 1967.

[32] S.N. Roy : Some Aspects of Multivariate Analysis. John Wiley and Sons, N.Y. 1957.

[33] S.N.Roy and R.E.Bargmann : Tests of Multiple Independence and the Associated Confidence Bounds. Annals of Mathematical Statistics, 1958, 29, 491-503.

[34] C.A. Smith : Linear Function Maximizing Intraclass Correlation. Annals Eugenics, Camg. 1953, 101, 286-292.

[35] R.G.D. Steel : Minimum Generalized Variance for a Group of Linear Function. Ph.D. Dissertation, Iowa State College, Ames, Iowa. 1949.

[36] ____________ : Minimum Generalized Variance for a Set of Linear Function. Annals of Mathematical Statistics, 1957, 41, 456-460.

[37] S.S. Stevens : On the Theory of Scales of Measurement. Science, 1946, 103, 670-680.

[38] L.L. Thurstone : Multiple Factor Analysis. University of Chicago Press, Chicago. 1947.

[39] A. Wald and R.J. Brookner : On the Distribution of Wilks' Statistic for Testing the Independence of Several Groups of Variates. Annals of Mathematical Statistics, 1947, 12, 137-152.

[40] S.S. Wilks : Certain Generalization in the Analysis of Variance. Biometrika, 1932, 24.

[41] ________ : Weighing System for Linear Function of Correlated Variables When There Is No Dependence Variables. Psychometrika, 1938, 3, 23-40.

[42] ________ : On the Independence of k sets of Normally Distributed Statistical Variables. Econometrika, 1953, 3, 309-326.

[42] ________ : The Likelihood Test of Independence in Contingency Table. Annals of Mathematical Statistics, 1956, 6, 190-196.

APPENDIX A1

CONTROL CARDS FOR THE COMPUTER PROGRAM

The control cards for the computer program as follow:

1st card: in FORMAT(6I2)

Col. 1- 2 ND1- No. of levels of the 1st response variable.

Col. 3- 4 ND2- No. of levels of the 2nd response variable.

Col. 5- 6 ND3- No. of levels of the 3rd response variable.

Col. 7- 8 ND4- No. of levels of the 4th response variable.

Col. 9-10 ND5- No. of levels of the 5th response variable.

Col.11-12 NVAR- No. of sets (response variables).

****Note: The omitted response variable is set to 1.**

2nd card: in FORMAT(4I2, 5I4)

Col. 1- 2 IDX1- Level of the 1st factor (row number).

Col. 3- 4 IDX3- Level of the 3rd factor.

Col. 5- 6 IDX4- Level of the 4th factor.

Col. 7- 8 IDX5- Level of the 5th factor.

Col. 9-28 Cell frequencies for the levels of the 2nd factor.

Col. 9-12 (IDX1, 1, IDX3, IDX4, IDX5)

Col.13-16 (IDX1, 2, IDX3, IDX4, IDX5)

Col.17-20 (IDX1, 3, IDX3, IDX4, IDX5)

Col.21-24 (IDX1, 4, IDX3, IDX4, IDX5)

Col.25-28 (IDX1, 5, IDX3, IDX4, IDX5)

***Note: This layout is chosen since most user will think of a contingency table as a two-way table of rows (factor 1) and column (factor 2). The levels of the other factors (3, 4, 5), if any are kept constant.

OUTPUT:

Write the intermediate information on TAPE 10, the temporary data set needed for the following computer program.

a. Initial estimated and approximated weights.

b. Fletcher and Powell descent method.

c. Canonical-partial and canonical-multiple correlations.

d. Reduction weights for the original E-matrix.

The following records are stored in TAPE 10:

1st record: in FORMAT(2I5, 2E20.6)

N- Number of variables (N=ND1+ND2+ND3+ND4+ND5-5).

LIMIT- Maximum number of iterations.

EST- Estimated minimum value of the given function.

EPS- Permissible error.

2nd record: in FORMAT 6I2)

NSET- Number of sets.

(NRST(I),I=1,NSET) for N1,N2, N3, N4, N5.

(N1=ND1-1, N2=ND2-1, N3=ND3-1, N4=ND4-1, N5=ND5-1)

3rd record: in FORMAT(4E20.6)

(X(I),I=1,N) the canonical weights

a. In C-E-R , the assumed canonical weights values will be stored.

b. In ICW, the assumed canonical weights will be replaced by the estimated or approximated weights values.

c. In FPM, the normalized canonical weights will replace the previous data.

OUTPUT:

Write the intermediate information on TAPE 10, the temporary data set needed for the following computer program.

a. Initial estimated and approximated weights.

b. Fletcher and Powell's descent method.

c. Canonical-partial and canonical-multiple correlations.

d. Reduction weights for the original E-matrix.

The Following records are stored in TAPE 10:

1st record: in FORMAT(2I5, 2E20.6)

N- Number of variables. (N=ND1+ND2+ND3+ND4+ND5-5).

LIMIT- Maximum number of iterations.

EST- Estimated minimum value of the given function.

EPS- Permissible error.

2nd record: in FORMAT(6I2)

NSET- Number of sets.

(NRST(I),I=1,NSET) for N1, N2, N3, N4, N5

(N1=ND1-1, N2=ND2-1, N3=ND3-1, N4=ND4-1, N5=ND5-1)

3rd record: in FORMAT(4E20.6)

(X(I),I=1,N) the categorical weights

4^{th} record: in FORMAT(5I8)

(NT(I,J),J-1,NRST(I)),I=1,NSET) the marginal totals for each set.

5^{th} record: in FORMAT(I8)

NTAL- The grand total of the contingency table.

6^{th} record: in FORMAT(2I2, (5E20.6))

ND,MK,((T(J,K),K=1,ND),J=1,MK) The conditional inverses of T_i, $i = 1, 2, 3, 4, 5$.

7^{th} record: in FORMAT(4E20.6) ... as many as needed.

All rank reduced submatrices of the super-matrix R, stored in column-wise, i.e. R_{11}, R_{12}, R_{22}, R_{13}, R_{23}, R_{33}, R_{14}, R_{24}, R_{34}, R_{44}, R_{15}, R_{25}, R_{35}, R_{45}, R_{55}.

APPENDIX A2

COMPUTER PROGRAM FOR CONTINGENCY TABLES

TO E TO R MATRIX

"C - E - R"

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C    THE TEMPORARY STORAGE IN TAPE
C
C
C TAPE 10- STORE THE FOLLOWING INFORMATION FOR THE FOLLOWING PROGRAMS :
C *******
C
C       1ST RECORD -
C           *STORED IN FORMAT(2I2,2E20.6)
C         N - NO. OF CATEGORICAL WEIGHTS.
C         LIMIT - MAXIMUM NO. OF ITERATIONS.
C         EST - ESTIMATED MINIMINUM OF A GIVEN FUNCTION.
C         EPS - PERMISSABLE ERROR.
C
C       2ND RECORD -
C           *STORED IN FORMAT(6I2)
C        NSET - NO. OF SETS.
C         (NRST(I),I=1,NSET) - FOR N1,N2,N3,N4,N5.
C
C       3RD RECORD -
C           *STORED IN FORMAT(4E20.6)
C         (X(I),I=1,N) - THE CATEGORICAL WEIGHTS FROM THE 'FLETCHER INITIA
C                          PROGRAM, AND WILL STORE THE RESULT FROM THIS PROG
C
C       4TH  RECORD
C           *STORED IN FORMAT(5I8)
C         (NT(I,J),J=1,NRST(I)), I=1,NSETS)
C         - THE MARGINAL TOTAL OF THE CONTINGENCY TABLE.
C
C       5TH  RECORD
C           *STORED IN FORMAT(I8)
C        NTAL - THE GRAND TOTAL OF THE CONTINGENCY TABLE.
C
C       6TH  RECORD -
C           *STORED IN FORMAT(2I2,(5E20.6))
C        NO,MK, (((T(I,J,K),K=1,NO),J=1,MK),I=1,NSETS)
C          - THE CONDITIONAL INVERSES.
C
C       7TH  RECORD ....
C
C           *STORED IN FORMAT(4E20.6)
C          ALL R MATRICES FROM THE 'CONTINGENCY TABLE TO E TO R' PROGRAM.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C THIS IS THE MAIN PROGRAM FOR DIRECT CALCULATION OF R-MATRIX.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C       THIS IS MAIN CALLING PROGRAM TO CALL ALL SUBROUTINE AND FUNCTION
C       SUBPROGRAMS TO  READ A CONTINGENCY TABLE UP TO 5 LEVELS INTO A SIN
C       ARRAY.
C       THE INPUT-CARD FORMAT IS AS FOLLOWING.
C          RESPONSE ON LEVEL 1 (ROW NO.) ON COLUMN 2.
C          RESPONSE ON LEVEL 3 (NO. ON 3RD LEVEL) ON COLUMN 4.
```

```
C        RESPONSE ON LEVEL 4 (NO. ON 4TH LEVEL) ON COLUMN 6.
C        RESPONSE ON LEVEL 5 (NO. ON 5TH LEVEL) ON COLUMN 8.
C     SET 4TH AND 5TH LEVELS TO 1. IF ONLY 3 VARIABLES.
C     SET 5TH LEVEL TO 1. IF ONLY 4 VARIABLES.
C
C     THE FREQUENCES WILL BE.
C        ACCORDING TO ROW-WISE DATA FROM THE CONTINGENCY TABLE(UP TO 5
C     ELEMENTS). COL(1) ON COLUMN 12. COL(2) ON COLUMN 16.COL(3) ON COLU
C     COL(4) ON COLUMN 24. COL(5) ON COLUMN 28.
C     NO1 IS NUMBER OF DISTINCT RESPONSE FROM 1ST LEVEL.
C     NO2 IS NUMBER OF DISTINCT RESPONSE FROM 2ND LEVEL.
C     NO3 IS NUMBER OF DISTINCT RESPONSE FROM 3RD LEVEL.
C     NO4 IS NUMBER OF DISTINCT RESPONSE FROM 4TH LEVEL.
C     NO5 IS NUMBER OF DISTINCT RESPONSE FROM 5TH LEVEL.
C     THE OMITTED LEVEL IS ALWAYS SET TO 1.
C     NVAR IS THE NUMBER OF LEVELS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      INTEGER OUPUT
      DIMENSION ISET(4),IFR(5)
      DIMENSION AR(3125)
      COMMON EE(15.5),NT(5.5)
      COMMON NO1.NO2.NO3.NO4.NO5.NVAR.OUPUT.MK1.MK2.MK3.MK4.MK5.OZ.IEND.
     1E(15.5.5)
      REWIND 10
      INPUT = 5
      OUPUT = 6
      OZ=0.E0
      READ(INPUT.101) NO1.NO2.NO3.NO4.NO5,NVAR
  101 FORMAT(6I2)
      WRITE(OUPUT,4210)
 4210 FORMAT(1H1////T35.32HTHE CONTINGENCY TABLE INPUT DATA    )
      WRITE(IOUT.4201) NVAR.NO1.NO2.NO3.NO4.NO5
 4201  FORMAT(1HO///.T10.29HTHE NUMBER OF SETS (NSETS)  = .I6//T10,47HTH
     1E NUMBER OF RESPONSES OF FIRST LEVEL (NO1)  = .I6//T10,48HTHE NUMB
     2ER OF RESPONSES OF SECOND LEVEL (NO2) =  .I5//T10.47HTHE NUMBER OF
     3 RESPONSES OF THIRD LEVEL (NO3)  = .I6//T10.48HTHE NUMBER OF RESPO
     4NSES OF FOURTH LEVEL (NO4) = .I5//T10,47HTHE NUMBER OF RESPONSES O
     5F FIFTH LEVEL (NO5)  = .I6)
      NTAL=NO1*NO3*NO4*NO5
      WRITE(OUPUT.4205)
 4205 FORMAT(1HO//T15. 17HCONTINGENCY TABLE    )
      WRITE(OUPUT.4209) NO2
 4209 FORMAT(1HO.T10,5X,5HLEVEL.9X,5HLEVEL,9X,5HLEVEL.3X,5HLEVEL.7X,5HLE
     1VEL /T10,6X,3H(1),5X,3H(3),5X.3H(4).5X,3H(5). 9X,3H(2),1X,5H(1....
     2I2,1H) /)
      WRITE(OUPUT,4214)
 4214 FORMAT(1H .T10.100H**************************************************
     1**************************************************//)
      DO 10 I=1,NTAL
      READ(INPUT,102) (ISET(IA),IA=1.4),(IFR(IB),IB=1.NO2)
  102 FORMAT(4I2.5I4)
```

```
      WRITE(OUPUT,4200) (ISET(I),I=1,4),(IFR(IB),IB=1,ND2)
 4200 FORMAT(1H0,T10,4I8,5I12)
      DO 10 J=1,ND2
      LX=LT(ISET(1),J,ISET(2),ISET(3),ISET(4))
      AR(LX)=IFR(J)
   10 CONTINUE
      NNTAL=NTAL*ND2
      WRITE(OUPUT,1100) (AR(I),I=1,NNTAL)
 1100 FORMAT(1H1,T10,16HTHE SINGLE ARRAY  //(T10,5E20.6))
      CALL PARTA(AR,E,IEND)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THE PARTA SUBROUTINE IS TO PLACE THE SINGLE ARRAY DATA AR INTO THE
C     MATRIX, AND MANIPULATE TO FIND THE VARIANCE-COVARIANCE MATRICES.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
 1101 CALL LINKA
 1200 CALL EXIT
      END
```

```
      SUBROUTINE PARTA(AR,E,ILND)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THIS PROGRAM IS TO PLACE THE SINGLE ARRAY AR INTO E MATRIX WHICH I
C      THE FORM OF E(K,IB,IC).
C      K=1.   E11.  K=2      E12.  K=3.       E22.  K=4       E13.  K=5
C     K=6      E33.  K=7      E14.  K=8       E24.  K=9       E34.  K=10
C     K=11     E15.  K=12     E25.  K=13      E35.  K=14      E45.  K=15
C     AND THEN THE VARIANCE-COVARIANCE MATRICES ARE CALCULATED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      INTEGER OUPUT
      DIMENSION NDD(5),IX(5),AR(3125),E(15,5,5),RSUM(5),CSUM(5)
      COMMON EE(15,5),NT(5,5)
      COMMON ND1,ND2,ND3,ND4,ND5,NVAR,OUPUT,MK1,MK2,MK3,MK4,MK5,DZ
      IEND=15
      IF(NVAR-2) 10,11,10
   10 IF(NVAR-3)12,13,12
   12 IF(NVAR-4) 14,15,14
   11 IEND=3
      GO TO 16
   13 IEND=6
      GO TO 16
   15 IEND=10
   16 CONTINUE
   14 CONTINUE
      NDD(1)=ND1
      NDD(2)=ND2
      NDD(3)=ND3
      NDD(4)=ND4
      NDD(5)=ND5
      DO 17 K=1,IEND
      DO 17 I=1,5
      DO 17 J=1,5
   17 E(K,I,J)=DZ
      K=0
      DO 20 JA=1,NVAR
      DO 20 JB=1,JA
      K=K+1
      I1=NDD(JA)
      DO 30 IA=1,I1
      I2=NDD(JB)
      DO 31 IB=1,I2
      CALL MAT1(JA,JB,NDD,IX)
      I3=IX(1)
      I4=IX(2)
      I5=IX(3)
      I6=IX(4)
      DO 31 IC=1,I3
      DO 31 ID=1,I4
      DO 31 IE=1,I5
      DO 35 IG=1,I6
      IF(JA-JB) 36,37,38
   37 GO TO (98,36,98,36,36,98,36,36,36,98,36,36,36,36,98),K
```

```
 98 CALL MAT2(IA.IB.IC.ID.IE.IG.E.AR.K)
 35 CONTINUE
 36 GO TO (99.97.99.97.97.99.97.97.97,99.97.97.97.97.99).K
 99 CONTINUE
    GO TO 91
 97 CONTINUE
    CALL MAT3(IA.IB.IC.ID.IE.IG.E.AR.K)
 31 CONTINUE
    IF(JA-JB) 30.38.30
 30 CONTINUE
 38 CONTINUE
    GO TO (90.91.90.91.91.90.91.91.91,90.91.91.91.91.90).K
 90 CONTINUE
    CALL MAT5(I2.I1.E.DZ.RSUM.K)
    CALL MAT6(I2.I1.DZ.E.RSUM.CSUM.K)
    GO TO 89
 91 CALL MAT4(I2,I1.E.DZ.K.CSUM.RSUM)
    CALL PLACE(I2.I1.CSUM.RSUM.NT.JA.JB)
 89 CONTINUE
 20 CONTINUE
    CALL JDEX(NOD.NT.OUPUT.NVAR)
    WRITE(OUPUT.104)
104 FORMAT(1H1)
    CALL OUPTA(E)
    RETURN
    END
```

```
C    TO PLACE THE MARGINAL TOTAL IN AN ARRAY AND FIND THE GRAND TOTAL
      SUBROUTINE JOEX (NOD,NT,IOUT,NVAR)
      DIMENSION NOD(5),NT(5,5)
      INPUT = 10
      DO 10 JA=1,NVAR
      IA=NOD(JA)
      WRITE(IOUT,102) JA
 102  FORMAT(///T10,25HTHE MARGINAL TOTAL OF SET  ,I3)
      WRITE(IOUT,101) (NT(JA,J),J=1,IA)
 101  FORMAT(1H0,T10,5I10)
      WRITE(INPUT,103) (NT(JA,J),J=1,IA)
 103  FORMAT(5I8)
 10   CONTINUE
      NTAL = 0
      JB = NOD(1)
      DO 12 I=1,JB
 12   NTAL = NTAL + NT(1,I)
      WRITE(IOUT,105) NTAL
 105  FORMAT(1H0,T10,17HTHE GRAND TATOL = ,I10)
      WRITE(INPUT,103) NTAL
      RETURN
      END
```

```
      SUBROUTINE PLACE(I2,I3.CSUM.RSUM.NT.JA.JB)
      DIMENSION CSUM(5).RSUM(5).NT(5.5)
      DO 11 KB=1.I3
11    NT(JA.KB) = CSUM(KB)
      DO 14 MB=1.I2
14    NT(JB.MB) = RSUM(MB)
10    CONTINUE
      RETURN
      END



      SUBROUTINE MAT1(JA.JB.NOD.IX)
      DIMENSION NOD(5).IX(5)
      MX=1
      DO 41 KX=1.5
      IF(KX-JA) 42.41.42
   42 IF(KX-JB) 43.41.43
   43 IX(MX)=NOD(KX)
      MX=MX+1
   41 CONTINUE
      RETURN
      END



      SUBROUTINE MAT2(IA.IB.IC.ID.IE.IG.E.AR.K)
      INTEGER OUPUT
      DIMENSION AR(3125).E(15.5.5)
      COMMON EE(15.5).NT(5.5)
      COMMON NO1.NO2.NO3.NO4.NO5.NVAR.OUPUT.MK1.MK2.MK3.MK4.MK5.DZ
      GO TO(61.99.63.99.99.66.99.99.99.70.99.99.99.99.75).K
   61 LX=LT(IB.IC.ID.IE.IG)
      GO TO 80
   63 LX=LT(IC.IB.ID.IE.IG)
      GO TO 80
   66 LX=LT(IC.ID.IB.IE.IG)
      GO TO 80
   70 LX=LT(IC.ID.IE.IB.IG)
      GO TO 80
   75 LX=LT(IC.ID.IE.IG.IB)
   80 E(K.IB.IB)=E(K.IB.IB)+AR(LX)
   99 RETURN
      END
```

```
      SUBROUTINE MAT3(IA.IB.IC.ID.IE.IO.E.AR.K)
      INTEGER OUPUT
      DIMENSION AR(3125).E(15.5.5)
      COMMON EE(15.5).NT(5.5)
      COMMON ND1.ND2.ND3.ND4.ND5.NVAR.OUPUT.MK1.MK2.MK3.MK4.MK5.DZ
      GO TO (99.62.99.64.65.99.67.68.69.99.71.72.73.74.99).K
   62 LX=LT(IB.IA.IC.ID.IE)
      GO TO 76
   64 LX=LT(IB.IC.IA.ID.IE)
      GO TO 76
   65 LX=LT(IC.IB.IA.ID.IE)
      GO TO 76
   67 LX=LT(IB.IC.ID.IA.IE)
      GO TO 76
   68 LX=LT(IC.IB.ID.IA.IE)
      GO TO 76
   69 LX=LT(IC.ID.IB.IA.IE)
      GO TO 76
   71 LX=LT(IB.IC.ID.IE.IA)
      GO TO 76
   72 LX=LT(IC.IB.ID.IE.IA)
      GO TO 76
   73 LX=LT(IC.ID.IB.IE.IA)
      GO TO 76
   74 LX=LT(IC.ID.IE.IB.IA)
   76 E(K.IB.IA)=E(K.IB.IA)+AR(LX)
   99 RETURN
      END
```

```
      SUBROUTINE MAT4(I2,I3.E.DZ.K.CSUM.RSUM)
      INTEGER OUPUT
      DIMENSION E(15.5.5).CSUM(5).RSUM(5)
      OUPUT = 6
      DO 51 MB=1.I2
      RSUM(MB)=DZ
      DO 51 KB=1.I3
   51 RSUM(MB)=RSUM(MB)+E(K.MB.KB)
      DO 52 KB=1.I3
      CSUM(KB)=DZ
      DO 52 MB=1.I2
   52 CSUM(KB)=CSUM(KB)+E(K.MB.KB)
      AN=DZ
      DO 53 KB=1.I3
   53 AN=AN+CSUM(KB)
      DO 50 MB=1.I2
      DO 50 KB=1.I3
   50 E(K.MB.KB)=E(K.MB.KB)-RSUM(MB)*CSUM(KB)/AN
      RETURN
      END


      SUBROUTINE MAT5(I2.I3.E.DZ.RSUM.K)
      DIMENSION E(15.5.5).RSUM(5)
      DO 51 MB=1.I2
      RSUM(MB)=DZ
      DO 51 KB=1.I3
   51 RSUM(MB)=RSUM(MB)+E(K.MB.KB)
      RETURN
      END
```

```
      SUBROUTINE MAT6(I2,I3,DZ,E,RSUM,CSUM,K)
      DIMENSION E(15,5,5),RSUM(5),CSUM(5)
      COMMON EE(15,5)
      DO 52 MB=1,I2
      DO 52 KB=1,I3
   52 E(K,MB,KB)=DZ
      AN=DZ
      DO 53 MB=1,I2
   53 AN=AN+RSUM(MB)
      DO 54 I=1,I2
      E(K,I,I)=RSUM(I)
      EE(K,I)=RSUM(I)
   54 CSUM(I)=RSUM(I)
      DO 50 MB=1,I2
      DO 50 KB=1,I3
   50 E(K,MB,KB)=E(K,MB,KB)-RSUM(MB)*RSUM(KB)/AN
      RETURN
      END
```

```
      SUBROUTINE LINKA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THE SUBROUTINE LINKA IS TO FIND THE T INVERSE  FOR CALCULATE THE
C     R MATRICES.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      INTEGER OUPUT
      DIMENSION RFX11(5.5).RFX12(5.5).RFX21(5.5).RFX22(5.5).RFX31(5.5).
     1RFX32(5.5).RFX41(5.5).RFX42(5.5).RFX51(5.5).RFX52(5.5).EA(5).RF1(5
     2.5)
      DIMENSION ITEZ(6).WINIT(20)
      COMMON EE(15.5).NT(5.5)
      COMMON NO1.NO2.NO3.NO4.NO5.NVAR.OUPUT.MK1.MK2.MK3.MK4.MK5.DZ.IEND.
     1E(15.5.5)
      INPUT = 10
      REWIND 10
      WRITE(OUPUT.1013)
 1013 FORMAT(1H1/////T25.27HINFORMATION STORED IN TAPE    )
      NWTS = NO1 + NO2 + NO3 + NO4 + NO5 - 5
      NWT1 = 150
      WT2 = 0.E0
      WT3 = 1.E-5
      ITEZ(1) = NVAR
      ITEZ(2) = NO1- 1
      ITEZ(3) = NO2 - 1
      ITEZ(4) = NO3 - 1
      ITEZ(5) = NO4 - 1
      ITEZ(6) = NO5 - 1
      DO 1331 I=1.NWTS
 1331 WINIT(I) = 0.5E0
      DO 1304 JA=1.NVAR
      IA = ITEZ(JA+1) + 1
      WRITE(OUPUT.1305) JA.(NT(JA.I).I=1.IA)
 1305 FORMAT(1H0.T10.26HTHE MARGINAL TOTAL FOR SET .I3//(T10.5I8))
      WRITE(INPUT.1306) (NT(JA.I).I=1.IA)
 1306 FORMAT(5I8)
 1304 CONTINUE
      NTAL = 0
      IA = ITEZ(2) + 1
      DO 1308 I=1.IA
 1308 NTAL = NTAL + NT(1.I)
      WRITE(OUPUT.1309) NTAL
 1309 FORMAT(1H0//T10.17HTHE GRAND TOTAL = .I8)
      WRITE(INPUT.1306) NTAL
      IK=1
      K=0
      DO 40 I=1.IEND
      K=K+IK
      GO TO (41.40.41.40.40.41.40.40.40.41.40.40.40.40.41).K
   41 CONTINUE
      CALL PARTE(IEND.K.EA.NO)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
```

```
C     THE SUBROUTINE PARTE IS TO FIND THE EA VECTOR IN ORDER TO BE USED
C     CALCULTING THE T CONDITIONAL INVERSE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      CALL TPRM1(EA.NO.RF1.MK.OZ)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THE SUBROUTINE TPRM1 IS TO FIND THE T CONDITIONAL INVERSE DIRECTLY
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(K-1) 62.63.62
   63 CALL TPRM2(RF1.RFX11.RFX12.MK.NO.MK1.OUPUT)
      GO TO 40
   62 IF(K-3) 64.65.64
   65 CALL TPRM2(RF1.RFX21.RFX22.MK.NO.MK2.OUPUT)
      GO TO 40
   64 IF(K-6) 66.67.66
   67 CALL TPRM2(RF1.RFX31.RFX32.MK.NO.MK3.OUPUT)
      GO TO 40
   66 IF(K-10) 68.69.68
   69 CALL TPRM2(RF1.RFX41.RFX42.MK.NO.MK4.OUPUT)
      GO TO 40
   68 IF(K-15) 40.71.40
   71 CALL TPRM2(RF1.RFX51.RFX52.MK.NO.MK5.OUPUT)
   40 CONTINUE
      CALL PARTC(E.RFX11.RFX12.RFX21.RFX22.RFX31.RFX32.RFX41.RFX42.RFX51
     1.RFX52.IEND)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THE SUBROUTINE PARTC IS TO FIND THE FINAL OUTPUT OF THE R MATRICES
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      RETURN
      END
```

```
      SUBROUTINE PARTC(E,RFX11,RFX12,RFX21,RFX22,RFX31,RFX32,RFX41,RFX42
     1,RFX51,RFX52,IEND)
      INTEGER OUPUT
      DIMENSION E(15,5,5),RFX11(5,5),RFX12(5,5),RFX21(5,5),RFX22(5,5),RF
     1X31(5,5),RFX32(5,5),RFX41(5,5),RFX42(5,5),RFX51(5,5),RFX52(5,5)
      COMMON EE(15,5),NT(5,5)
      COMMON ND1,ND2,ND3,ND4,ND5,NVAR,OUPUT,MK1,MK2,MK3,MK4,MK5,DZ
      DZ = 0.E0
      WRITE(OUPUT,101)
  101 FORMAT(1H1)
      WRITE(OUPUT,102)
  102 FORMAT(1H0,T10,14HTHE R11 MATRIX )
      K=1
      CALL OUPT3(MK1,OUPUT,DZ)
      IF(IEND-1) 11,20,11
   11 CONTINUE
      WRITE(OUPUT,103)
  103 FORMAT(1H0,T10,14HTHE R12 MATRIX )
      K=2
      CALL OUPT2(ND1,ND2,MK1,MK2,RFX11,RFX22,E,DZ,K,OUPUT)
      K=3
      WRITE(OUPUT,104)
  104 FORMAT(1H0,T10,14HTHE R22 MATRIX )
      CALL OUPT3(MK2,OUPUT,DZ)
      IF(IEND-3) 12,20,12
   12 CONTINUE
      K=4
      WRITE(OUPUT,105)
  105 FORMAT(1H0,T10,14HTHE R13 MATRIX )
      CALL OUPT2(ND1,ND3,MK1,MK3,RFX11,RFX32,E,DZ,K,OUPUT)
      K=5
      WRITE(OUPUT,106)
  106 FORMAT(1H0,T10,14HTHE R23 MATRIX )
      CALL OUPT2(ND2,ND3,MK2,MK3,RFX21,RFX32,E,DZ,K,OUPUT)
      K=6
      WRITE(OUPUT,107)
  107 FORMAT(1H0,T10,14HTHE R33 MATRIX )
      CALL OUPT3(MK3,OUPUT,DZ)
      IF(IEND-6) 13,20,13
   13 CONTINUE
      K=7
      WRITE(OUPUT,108)
  108 FORMAT(1H0,T10,14HTHE R14 MATRIX )
      CALL OUPT2(ND1,ND4,MK1,MK4,RFX11,RFX42,E,DZ,K,OUPUT)
      K=8
      WRITE(OUPUT,109)
  109 FORMAT(1H0,T10,14HTHE R24 MATRIX )
      CALL OUPT2(ND2,ND4,MK2,MK4,RFX21,RFX42,E,DZ,K,OUPUT)
      K=9
      WRITE(OUPUT,110)
  110 FORMAT(1H0,T10,14HTHE R34 MATRIX )
      CALL OUPT2(ND3,ND4,MK3,MK4,RFX31,RFX42,E,DZ,K,OUPUT)
      K=10
      WRITE(OUPUT,111)
```

```
111 FORMAT(1H0,T10,14HTHE R44 MATRIX )
    CALL OUPT3(MK4,OUPUT,DZ)
    IF(IEND-10) 14,20,14
 14 CONTINUE
    K=11
    WRITE(OUPUT,112)
112 FORMAT(1H0,T10,14HTHE R15 MATRIX )
    CALL OUPT2(ND1,ND5,MK1,MK2,RFX11,RFX52,E,DZ,K,OUPUT)
    K=12
    WRITE(OUPUT,113)
113 FORMAT(1H0,T10,14HTHE R25 MATRIX )
    CALL OUPT2(ND2,ND5,MK2,MK5,RFX21,RFX52,E,DZ,K,OUPUT)
    K=13
    WRITE(OUPUT,114)
114 FORMAT(1H0,T10,14HTHE R35 MATRIX )
    CALL OUPT2(ND3,ND5,MK3,MK5,RFX31,RFX52,E,DZ,K,OUPUT)
    K=14
    WRITE(OUPUT,115)
115 FORMAT(1H0,T10,14HTHE R45 MATRIX )
    CALL OUPT2(ND4,ND5,MK4,MK5,RFX41,RFX52,E,DZ,K,OUPUT)
    K=15
    WRITE(OUPUT,116)
116 FORMAT(1H0,T10,14HTHE R55 MATRIX )
    CALL OUPT3(MK5,OUPUT,DZ)
 20 CONTINUE
    REWIND 10
    RETURN
    END
```

```
      SUBROUTINE PARTE(IEND.K.EA.ND)
      DIMENSION EA(5)
      COMMON EE(15.5).NT(5.5)
      COMMON ND1.ND2.ND3.ND4.ND5.NVAR.OUPUT.MK1.MK2.MK3.MK4.MK5.DZ
      ND=ND1
      IF(K-1) 43.70.43
   43 ND=ND2
      IF(K-3) 44.70.44
   44 ND=ND3
      IF(K-6) 45.70.45
   45 ND=ND4
      IF(K-10) 46.70.46
   46 ND=ND5
   70 CONTINUE
      DO 51 I=1.ND
   51 EA(I)=EE(K.I)
      RETURN
      END
```

```
      SUBROUTINE TPRM1(EA,NO,RF1,MK,DZ)
      DIMENSION EA(5),RF1(5,5)
      MK=NO-1
      TOTAL=DZ
      DO 10 I=1,NO
   10 TOTAL=TOTAL+EA(I)
      DO 11 I=1,5
      DO 11 J=1,5
   11 RF1(I,J)=DZ
      RF1(1,1)=SQRT(TOTAL/(EA(1)*(TOTAL-EA(1))))
      DO 12 I=2,MK
      TN1=TOTAL
      II=I
      III=I-1
      DO 13 IK=1,III
   13 TN1=TN1-EA(IK)
      TN2=TN1-EA(II)
      DO 12 J=1,I
      IF(I-J) 30,31,30
   30 RF1(I,J)=SQRT(EA(I)/(TN1*TN2))
      GO TO 12
   31 RF1(I,J)=SQRT(TN1/(EA(I)*TN2))
   12 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TPRM2(RF1.RX1.RX2.MK.ND.MKL.OUPUT)
      INTEGER OUPUT
      DIMENSION RF1(5.5).RX1(5.5).RX2(5.5)
      INPUT = 10
      MKL=MK
      DO 10 I=1.MK
      DO 10 J=1.ND
      RX1(I.J)=RF1(I.J)
   10 RX2(J.I)=RF1(I.J)
      WRITE(OUPUT.101) (( RX1(I.J).J=1.ND).I=1.MK)
  101 FORMAT(1H0.T10.25HTHE T CONDITIONAL INVERSE  //(T10.5E20.6))
      WRITE(INPUT.102) ND.MK.((RX1(I.J).J=1.ND).I=1.MK)
  102 FORMAT(2I2.(5E20.6))
      RETURN
      END
```

```
      SUBROUTINE OUPT2(N1,N2,M1,M2,R1,R2,E,DZ,K,OUPUT)
      INTEGER OUPUT
      DIMENSION R1(5,5),R2(5,5),E(15,5,5),W(5,5),WORK(5,5)
      INPUT = 10
      DO 10 I=1,5
      DO 10 J=1,5
      W(I,J)=DZ
   10 WORK(I,J)=DZ
      DO 11 I=1,M1
      DO 11 J=1,N2
      DO 11 L=1,N1
   11 WORK(I,J)= WORK(I,J)+R1(I,L)*E(K,L,J)
      DO 12 I=1,M1
      DO 12 J=1,M2
      DO 12 L=1,N2
   12 W(I,J)=W(I,J)+WORK(I,L)*R2(L,J)
      WRITE(OUPUT,101) ((W(I,J),J=1,M2),I=1,M1)
  101 FORMAT(1H0,T10,5E20.6)
      WRITE(INPUT,102) ((W(I,J),J=1,M2),I=1,M1)
  102 FORMAT(4E20.6)
      RETURN
      END
```

```
      SUBROUTINE OUPT3(N1,OUPUT,OZ)
      INTEGER OUPUT
      DIMENSION R(5,5)
      INPUT = 10
      DO 55 I=1,5
      DO 55 J=1,5
   55 R(I,J)=0.E0
      DO 10 I=1,5
   10 R(I,I)=1.E0
      WRITE(OUPUT,101) ((R(I,J),J=1,N1),I=1,N1)
  101 FORMAT(1H0,T10,5E20.6)
      WRITE(INPUT,102) ((R(I,J),J=1,N1),I=1,N1)
 102  FORMAT(4E20.6)
      RETURN
      END
```

```
      FUNCTION LT(I,J,K,L,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THIS FUNCTION SUBPROGRAM IS TO FIND THE INDEX FOR THE SINGLE ARRAY
C     FROM THE CONTINGENCY TABLE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      INTEGER OUPUT
      COMMON EE(15,5),NT(5,5)
      COMMON ND1,ND2,ND3,ND4,ND5,NVAR,OUPUT,MK1,MK2,MK3,MK4,MK5,DZ
      LT=((((M-1)*ND4+(L-1))*ND3+(K-1))*ND2+(J-1))*ND1+I
      RETURN
      END
```

```
      SUBROUTINE OUPTA(E)
      INTERGER OUPUT
      DIMENSION NOD(5),E(15,5,5)
      COMMON EE(15,5),NT(5,5)
      COMMON ND1,ND2,ND3,ND4,ND5,NVAR,OUPUT,MK1,MK2,MK3,MK4,MK5,DZ
      NOD(1)=ND1
      NOD(2)=ND2
      NOD(3)=ND3
      NOD(4)=ND4
      NOD(5)=ND5
      K=0
      DO 18 JA=1,NVAR
      DO 18 JB=1,JA
      K=K+1
      WRITE(OUPUT,104) JB,JA
 104  FORMAT(1H0,T10,5HTHE E,2I2,7H MATRIX  )
      I2=NOD(JB)
      I3 = NOD(JA)
      DO 18 IB=1,I2
      WRITE(OUPUT,105) (E(K,IB,IC),IC=1,I3)
 105  FORMAT(//(1X,T15,5E20.6))
   18 CONTINUE
      RETURN
      END
```

APPENDIX B

COMPUTER PROGRAM FOR INITIAL WEIGHTS
ESTIMATION AND APPROXIMATION
"I C W"

```
C
C
      DIMENSION RR(15.5.5).A(25).R(25).NRST(5).RHS(5.4).RGN(5.5)
      DIMENSION VAV(5).U1(5).U2(5).U3(5).U4(5).U5(5).V1(5).V2(5).V3(5).V
     14(5).V5(5).W1(5).W2(5).W3(5).W4(5).W5(5).R1(5).R2(5).R3(5).R4(5).R
     25(5)
      DIMENSION RS12(5.5).RS13(5.5).RS14(5.5).RS15(5.5).RS23(5.5).RS24(5
     1.5).RS25(5.5).RS34(5.5).RS35(5.5).RS45(5.5)
     1 X(25)
      COMMON ATOTL
      COMMON N1.N2.N3.N4.N5.NSETS.INPUT.IOUT.KSET
      DATA U1.U2.U3.U4.U5.V1.V2.V3.V4.V5.W1.W2.W3.W4.W5.R1.R2.R3.R4.R5/1
     100*0.E0/
      DATA NRST/5*0/
      REWIND 10
      INPUT = 10
      IOUT = 6
      LIMIT = 0
      DZ = 0.E0
      NTOTL = 0
      READ(INPUT.100) NWTS.NWT1.WT2.WT3
100   FORMAT(2I5.2E20.6)
      READ(INPUT.101) NSETS.(NRST(I).I=1.5)
101   FORMAT(6I2)
      DO 7 I=1.5
7     NTOTL = NTOTL + NRST(I)
      ATOTL = NTOTL
      KSET = 3
      IF(NSETS - 2) 2.1.2
2     KSET = 6
      IF(NSETS - 3) 3.1.3
3     KSET = 10
      IF (NSETS - 4) 4.1.4
4     KSET = 15
1     CONTINUE
      READ(INPUT.105) (A(I).I=1.NWTS)
105   FORMAT(4E20.6)
      N1 = NRST(1)
      N2 = NRST(2)
      N3 = NRST(3)
      N4 = NRST(4)
      N5 = NRST(5)
      DO 3311 JA=1.NSETS
      JA = NRST(JA)
3311  READ(INPUT.2012) (NT(JA.I).I=1.IA)
2012  FORMAT(5I8)
      READ(INPUT.2012) NTAL
      DO 3312 JA=1.NSETS
3312  READ(INPUT.2013) ND.MK.((RS12(I.J).J=1.ND).I=1.MK)
2013  FORMAT(2I2.(5E20.6))
      CALL REAIN(RR.NSETS.INPUT.NRST.KSET)
5     CONTINUE
2000  LIMIT = LIMIT + 1
      ISTEP = 1
```

```
      GO TO (3101,3102),LIMIT
 3101 CONTINUE
      CALL RIOUT(RR,NSETS,IOUT,NRST,KSET)
      GO TO 3103
 3102 CONTINUE
      CALL ROUT(RR,NSETS,IOUT,NRST,KSET)
 3103 CONTINUE
      CALL EMPTY(RS12,RS13,RS14,RS15,RS23,RS24,RS25,RS34,RS35,RS45,DZ)
      GO TO (3201,3202),LIMIT
 3201 CALL RS1(RS12,RS13,RS14,RS15,RS23,RS24,RS25,RS34,RS35,RS45,RR,N1,N
     12,N3,N4,N5,IOUT,NSETS)
      GO TO 3203
 3202 CALL RS2(RS12,RS13,RS14,RS15,RS23,RS24,RS25,RS34,RS35,RS45,RR,N1,N
     12,N3,N4,N5,IOUT,NSETS)
 3203 CONTINUE
      GO TO (3301,3302),LIMIT
 3301 N = N1
      GO TO 3303
 3302 N = N2
 3303 CONTINUE
      DO 19 I=1,N
      DO 19 J=1,N
 19   RSN(I,J) = RS12(I,J)
      LM = 0
 9000 DO 17 I=1,N
      DO 17 J=I,N
      IJ = (J*J - J)/2 + I
 17   A(IJ) = RSN(I,J)
      NNN = (N - 1)*N/2 + N
      WRITE(IOUT,202) (A(I),I=1,NNN)
 202  FORMAT(1H0,T10,17HTHE PACKED MATRIX   //(T10,5E20.6))
      CALL EIGEN(A,R,N,0)
      WRITE(IOUT,203) A(1)
 203  FORMAT(1H0,T10,25HTHE LARGEST EIGEN-VALUE  //(T10,5E20.6))
      WRITE(IOUT,204) (R(I),I=1,N)
 204  FORMAT(1H0,T10,16HTHE EIGEN-VECTOR  //(T10,5E20.6))
      IF(LIMIT - 1) 1000,1001,1000
 1001 CALL SECOD(U1,V1,V2,W1,W2,W3,R1,R2,R3,R4,R,ISTEP,N,RHS,SQRT(A(1)))
      GO TO 566
 1000 CALL SECOD(U2,V3,U3,W4,U4,V4,R5,U5,V5,W5,R,ISTEP,N,RHS,SQRT(A(1)))
 566  IF(NSETS - 2) 25,25,26
 26   CONTINUE
      CALL RLOAD(RS12,RS13,RS14,RS15,RS23,RS24,RS25,RS34,RS35,RS45,RSN,N
     1,N1,N2,N3,N4,N5,LIMIT,ISTEP,LM,NSETS,IGO)
      GOTO (9001,9000),IGO
 9001 CONTINUE
 25   CONTINUE
      GO TO (2000,2004),LIMIT
 2004 CONTINUE
      NS = NSETS - 1
      IKX = 1
      KN = 0
      CALL FINAL(VAV,U1,V1,N1,R1,NS,N1,IOUT,RHS,IKX)
      CALL DORMF(VAV,N1,IOUT,KN,X)
```

```
      IKX = 2
      KN = KN + N1
      CALL FINAL(VAV,U2,V2,W2,R2,NS,N2,IOUT,RHS,IKX)
      CALL DORMF(VAV,N2,IOUT,KN,X)
      IF(NSETS - 2) 98,99,98
 98   CONTINUE
      IKX = 3
      KN = KN + N2
      CALL FINAL(VAV,U3,V3,W3,R3,NS,N3,IOUT,RHS,IKX)
      CALL DORMF(VAV,N3,IOUT,KN,X)
      IF(NSETS - 3) 97,99,97
 97   CONTINUE
      IKX = 4
      KN = KN + N3
      CALL FINAL(VAV,U4,V4,W4,R4,NS,N4,IOUT,RHS,IKX)
      CALL DORMF(VAV,N4,IOUT,KN,X)
      IF(NSETS - 4) 96,99,96
 96   CONTINUE
      IKX = 5
      KN = KN + N4
      CALL FINAL(VAV,U5,V5,W5,R5,NS,N5,IOUT,RHS,IKX)
      CALL DORMF(VAV,N5,IOUT,KN,X)
 99   CONTINUE
      CALL FILUP(X,NTOTL,INPUT)
      REWIND 10
      STOP
      END
      SUBROUTINE REAIN(R,NSETS,INPUT,NRST,KSET)
      DIMENSION R(15,5,5),NRST(5)
      K = 1
      DO 10 J=1,NSETS
      NA = NRST(J)
      DO 10 I=1,J
      NB = NRST(I)
      READ(INPUT,102) ((R(K,IX,JX),JX=1,NA),IX=1,NB)
 102  FORMAT(4E20.6)
      K = K + 1
      IF(K - KSET), 10,10,11
 10   CONTINUE
 11   CONTINUE
      RETURN
      END
C
```

```
      SUBROUTINE RIOUT(R,NSETS,IOUT,NRST,KSET)
      DIMENSION R(15,5,5),NRST(5)
      WRITE(IOUT,100)
100   FORMAT(1H1,T25,49HINITIAL ESTIMATED WEIGHTS FOR FLETCHER AND POWEL
     1L )
      WRITE(IOUT,608) NSETS,(NRST(I),I=1,5)
608   FORMAT(1H0,T10,28HTHE NUMBER OF SETS (NSETS) = ,I6//T10,37HTHE NUM
     1BER OF ROWS OF (1,1) SET N1  = ,I6//T10,37HTHE NUMBER OF ROWS OF (
     22,2) SET N2  = ,I6//T10,37HTHE NUMBER OF ROWS OF (3,3) SET N3  = ,
     3I6//T10,37HTHE NUMBER OF ROWS OF (4,4) SET N4  = ,I6//T10,37HTHE N
     4UMBER OF ROWS OF (5,5) SET N5  = ,I6)
      WRITE(IOUT,140)
140   FORMAT(1H0,T35,14HTHE INPUT DATA  )
      K = 1
      DO 10 J=1,NSETS
      NA = NRST(J)
      DO 10 I=1,J
      NB = NRST(I)
      WRITE(IOUT,130) ((R(K,IX,JX),JX=1,NA),IX=1,NB)
130   FORMAT(1H0//(T10,5E20.6))
      K = K + 1
      IF(K - KSET) 10,10,11
10    CONTINUE
11    CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE ROUT(R.NSETS.IOUT.NRST.KSET)
      DIMENSION R(15.5.5).NRST(5)
      WRITE(IOUT.140)
  140 FORMAT(1H1.T35.26HTHE TRANSPOSED INPUT DATA     )
      K = 1
      DO 10 J=1,NSETS
      NA = NRST(J)
      DO 10 I=1.J
      NB = NRST(I)
      WRITE(IOUT,130) ((R(K.IX.JX).JX=1.NA),IX=1,NB)
  130 FORMAT(1H0//(T10,5E20.6))
      K = K + 1
      IF(K - KSET) 10.10.11
  10  CONTINUE
  11  CONTINUE
  605 RETURN
      END
      SUBROUTINE EMPTY(RS12.RS13.RS14.RS15.RS23.RS24.RS25.RS34.RS35.RS45
     1,DZ)
      DIMENSION RS12(5.5).RS13(5.5).RS14(5.5).RS15(5.5).RS23(5.5).RS24(5
     1.5).RS25(5.5).RS34(5.5).RS35(5.5).RS45(5.5)
      DO 1002 I=1.5
      DO 1002 J=1.5
      RS12(I.J) = DZ
      RS13(I.J) = DZ
      RS14(I.J) = DZ
      RS15(I.J) = DZ
      RS23(I.J) = DZ
      RS24(I.J) = DZ
      RS25(I.J) = DZ
      RS34(I.J) = DZ
      RS35(I.J) = DZ
      RS45(I.J) = DZ
 1002 CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE RLOAD(RS12,RS13,RS14,RS15,RS23,RS24,RS25,RS34,RS35,RS45
     1,RSN,N,N1,N2,N3,N4,N5,LIMIT,ISTEP,LM,NSETS,IGO)
      DIMENSION RS12(5,5),RS13(5,5),RS14(5,5),RS15(5,5),RS23(5,5),RS24(5
     1,5),RS25(5,5),RS34(5,5),RS35(5,5),RS45(5,5),RSN(5,5)
      IGO = 2
26    LM = LM + 1
      IF(LM - 1) 28,27,28
27    CONTINUE
      GO TO (2501,2502),LIMIT
2501  N = N1
      GO TO 2601
2502  N = N3
2601  DO 29 I=1,N
      DO 29 J=1,N
29    RSN(I,J) = RS13(I,J)
      ISTEP = ISTEP + 1
      GO TO 9000
28    IF(LM - 2) 30,31,30
31    CONTINUE
      GO TO (2701,2702),LIMIT
2701  N = N2
      GO TO 2801
2702  N = N3
2801  DO 32 I=1,N
      DO 32 J=1,N
32    RSN(I,J) = RS23(I,J)
      ISTEP = ISTEP + 1
      GO TO 9000
30    IF(NSETS - 3) 25,25,434
434   IF(LM - 3) 34,35,34
35    CONTINUE
      GO TO (2703,2704),LIMIT
2703  N = N1
      GO TO 2802
2704  N = N4
2802  DO 36 I=1,N
      DO 36 J=1,N
36    RSN(I,J) = RS14(I,J)
      ISTEP = ISTEP + 1
      GO TO 9000
34    IF(LM - 4) 37,38,37
38    CONTINUE
      GO TO (2705,2706),LIMIT
2705  N = N2
      GO TO 2803
2706  N = N4
2803  DO 39 I=1,N
      DO 39 J=1,N
39    RSN(I,J) = RS24(I,J)
      ISTEP = ISTEP + 1
      GO TO 9000
37    IF(LM - 5) 51,40,51
40    GO TO (2711,2712),LIMIT
```

```
2711 N = N3
     GO TO 2804
2712 N = N4
2804 DO 41 I=1,N
     DO 41 J=1,N
41   RSN(I,J) = RS34(I,J)
     ISTEP = ISTEP + 1
     GO TO 9000
51   IF(NSETS - 4) 25,25,52
52   IF(LM - 6) 53,54,53
54   CONTINUE
     GO TO (2713,2714),LIMIT
2713 N = N1
     GO TO 2805
2714 N = N5
2805 DO 61 I=1,N
     DO 61 J=1,N
61   RSN(I,J) = RS15(I,J)
     ISTEP = ISTEP + 1
     GO TO 9000
53   IF(LM - 7) 56,55,56
55   CONTINUE
     GO TO (2715,2716),LIMIT
2715 N = N2
     GO TO 2806
2716 N = N5
2806 DO 62 I=1,N
     DO 62 J=1,N
62   RSN(I,J) = RS25(I,J)
     ISTEP = ISTEP + 1
     GO TO 9000
56   IF(LM - 8) 57,58,57
58   GO TO (2717,2718),LIMIT
2717 N = N3
     GO TO 2807
2718 N = N5
2807 DO 63 I=1,N
     DO 63 J=1,N
63   RSN(I,J) = RS35(I,J)
     ISTEP = ISTEP + 1
     GO TO 9000
57   IF(LM - 9) 25,59,25
59   GO TO (2719,2720),LIMIT
2719 N = N4
     GO TO 2808
2720 N = N5
2808 DO 64 I=1,N
     DO 64 J=1,N
64   RSN(I,J) = RS45(I,J)
     ISTEP = ISTEP + 1
9000 CONTINUE
     RETURN
25   CONTINUE
     IGO = 1
```

```
RETURN
END
```

```
C
      SUBROUTINE SECOD(V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,R,ISTEP,N,RHS,A)
      DIMENSION V1(5),V2(5),V3(5),V4(5),V5(5),V6(5),V7(5),V8(5),V9(5),V
     10(5),R(5),RHS(5,4)
      GO TO (1,2,3,4,5,6,7,8,9,10,11),ISTEP
1     DO 901 I=1,N
901   V1(I) = R(I)
      RHS(1,1) = A
      RHS(2,1) = A
      GO TO 11
2     DO 902 I=1,N
902   V2(I) = R(I)
      RHS(1,2) = A
      RHS(3,1) = A
      GO TO 11
3     DO 903 I=1,N
903   V3(I) = R(I)
      RHS(2,2) = A
      RHS(3,2) = A
      GO TO 11
4     DO 904 I=1,N
904   V4(I) = R(I)
      RHS(1,3) = A
      RHS(4,1) = A
      GO TO 11
5     DO 905 I=1,N
905   V5(I) = R(I)
      RHS(4,2) = A
      RHS(2,3) = A
      GO TO 11
6     DO 906 I=1,N
906   V6(I) = R(I)
      RHS(4,3) = A
      RHS(3,3) = A
      GO TO 11
7     DO 907 I=1,N
907   V7(I) = R(I)
      RHS(1,4) = A
      RHS(5,1) = A
      GO TO 11
8     DO 908 I=1,N
908   V8(I) = R(I)
      RHS(2,4) = A
      RHS(5,2) = A
      GO TO 11
9     DO 909 I=1,N
909   V9(I) = R(I)
      RHS(3,4) = A
      RHS(5,3) = A
      GO TO 11
10    DO 910 I=1,N
910   V10(I) = R(I)
      RHS(4,4) = A
      RHS(5,4) = A
```

```
11    RETURN
      END
```

```
C
      SUBROUTINE RS2(RS12,RS13,RS14,RS15,RS23,RS24,RS25,RS34,RS35,RS45,R
     1, N1,N2,N3,N4,N5,IOUT,NSETS)
      DIMENSION RS12(5,5),RS13(5,5),RS14(5,5),RS15(5,5),RS23(5,5),RS24(5
     1,5),RS25(5,5),RS34(5,5),RS35(5,5),RS45(5,5)
      DIMENSION R(15,5,5)
      WRITE(IOUT,100)
  100 FORMAT(1H0,T10,25HTHE TRANSPOSED R-STAR SET )
      KX = 2
      DO 10 I=1,N2
      DO 10 J=1,N2
      DO 10 K=1,N1
   10 RS12(I,J) = RS12(I,J) + R(KX,K,J)*R(KX,K,I)
      WRITE(IOUT,101) ((RS12(I,J),J=1,N2),I=1,N2)
  101 FORMAT(1H0//(T10,5E20.6))
      IF(NSETS - 2) 501,5,501
  501 CONTINUE
      KX = 4
      DO 11 I=1,N3
      DO 11 J=1,N3
      DO 11 K=1,N1
   11 RS13(I,J) = RS13(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 5
      DO 12 I=1,N3
      DO 12 J=1,N3
      DO 12 K=1,N2
   12 RS23(I,J) = RS23(I,J) + R(KX,K,J)*R(KX,K,I)
      WRITE(IOUT,101) ((RS13(I,J),J=1,N3),I=1,N3)
      WRITE(IOUT,101) ((RS23(I,J),J=1,N3),I=1,N3)
      IF(NSETS - 3) 1,5,1
    1 CONTINUE
      KX = 7
      DO 13 I=1,N4
      DO 13 J=1,N4
      DO 13 K=1,N1
   13 RS14(I,J) = RS14(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 8
      DO 14 I=1,N4
      DO 14 J=1,N4
      DO 14 K=1,N2
   14 RS24(I,J) = RS24(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 9
      DO 15 I=1,N4
      DO 15 J=1,N4
      DO 15 K=1,N3
   15 RS34(I,J) = RS34(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 11
      DO 16 I=1,N5
      DO 16 J=1,N5
      DO 16 K=1,N1
   16 RS15(I,J) = RS15(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 12
      DO 17 I=1,N5
      DO 17 J=1,N5
```

```
      DO 17 K=1,N2
17    RS25(I,J) = RS25(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 13
      DO 18 I=1,N5
      DO 18 J=1,N5
      DO 18 K=1,N3
18    RS35(I,J) = RS35(I,J) + R(KX,K,J)*R(KX,K,I)
      KX = 14
      DO 19 I=1,N5
      DO 19 J=1,N5
      DO 19 K=1,N4
19    RS45(I,J) = RS45(I,J) + R(KX,K,J)*R(KX,K,I)
      WRITE(IOUT,101) ((RS15(I,J),J=1,N5),I=1,N5)
      WRITE(IOUT,101) ((RS25(I,J),J=1,N5),I=1,N5
      WRITE(IOUT,101) ((RS35(I,J),J=1,N5),I=1,N5
      WRITE(IOUT,101) ((RS45(I,J),J=1,N5),I=1,N5
5     CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE RS1(FS12.RS13.RS14.RS15.RS23.RS24.RS25.RS34.RS35.RS45.
     1 R.N1.N2.N3.N4.N5.IOUT.NSET6)
      DIMENSION RS12(5.5).RS13(5.5).RS14(5.5).RS15(5.5).RS23(5.5).RS24(5
     1.5).RS25(5.5).RS34(5.5).RS35(5.5).RS45(5.5)
      DIMENSION R(15.5.5)
      WRITE(IOUT.100)
 100  FORMAT(1H0.T10.10HR-STAR SET )
      KX = 2
      DO 10 I=1.N1
      DO 10 J=1.N1
      DO 10 K=1.N2
 10   RS12(I.J) = RS12(I.J) + R(KX.I.K)*R(KX.J.K)
      WRITE(IOUT.101) ((RS12(I.J).J=1.N1).I=1.N1)
 101  FORMAT(1H0//(T10.5E20.6))
      IF(NSETS - 2) 501.5.501
 501  CONTINUE
      KX = 4
      DO 11 I=1.N1
      DO 11 J=1.N1
      DO 11 K=1.N3
 11   RS13(I.J) = RS13(I.J) + R(KX.I.K)*R(KX.J.K)
      KX = 5
      DO 12 I=1.N2
      DO 12 J=1.N2
      DO 12 K=1.N3
 12   RS23(I.J) = RS23(I.J) + R(KX.I.K)*R(KX.J.K)
      WRITE(IOUT.101) ((RS13(I.J).J=1.N1).I=1.N1)
      WRITE(IOUT.101) ((RS23(I.J).J=1.N2).I=1.N2)
      IF(NSETS - 3) 1.5.1
 1    CONTINUE
      KX = 7
      DO 13 I=1.N1
      DO 13 J=1.N1
      DO 13 K=1.N4
 13   RS14(I.J) = RS14(I.J) + R(KX.I.K)*R(KX.J.K)
      KX = 8
      DO 14 I=1.N2
      DO 14 J=1.N2
      DO 14 K=1.N4
 14   RS24(I.J) = RS24(I.J) + R(KX.I.K)*R(KX.J.K)
      KX = 9
      DO 15 I=1.N3
      DO 15 J=1.N3
      DO 15 X=1.N4
 15   RS34(I.J) = RS34(I.J) + R(KX.I.K)*R(KX.J.K)
      WRITE(IOUT.101) ((RS14(I.J,.J=1.N1).I=1.N1)
      WRITE(IOUT.101) ((RS24(I.J).J=1.N2).I=1.N2)
      WRITE(IOUT.101) ((RS34(I.J).J=1.N3).I=1.N3)
      IF(NSETS - 4) 2.5.2
 2    CONTINUE
      KX = 11
      DO 16 I=1.N1
      DO 16 J=1.N1
```

```
      DO 16 K=1,N5
16    RS15(I,J) = RS15(I,J) + R(KX,I,K)*R(KX,J,K)
      KX = 12
      DO 17 I=1,N2
      DO 17 J=1,N2
      DO 17 K=1,N5
17    RS25(I,J) = RS25(I,J) + R(KX,I,K)*R(KX,J,K)
      KX = 13
      DO 18 I=1,N3
      DO 18 J=1,N3
      DO 18 K=1,N5
18    RS35(I,J) = RS35(I,J) + R(KX,I,K)*R(KX,J,K)
      KX = 14
      DO 19 I=1,N4
      DO 19 J=1,N4
      DO 19 K=1,N5
19    RS45(I,J) = RS45(I,J) + R(KX,I,J)*R(KX,J,K)
      WRITE(IOUT,101) ((RS15(I,J),J=1,N1),I=1,N1)
      WRITE(IOUT,101) ((RS25(I,J),J=1,N2),I=1,N2)
      WRITE(IOUT,101) ((RS35(I,J),J=1,N3),I=1,N3)
      WRITE(IOUT,101) ((RS45(I,J),J=1,N4),I=1,N4)
5     CONTINUE
      RETURN
      END
C
```

```
C
      SUBROUTINE DORMF(VAV,N,IOUT,KN,X)
      DIMENSION VAV(5) ,X(25)
      SUM = 0.E0
      DO 1 I=1,N
 1    SUM = SUM + VAV(I)**2
      QSUM =  SQRT(SUM)
      DO 2 I=1,N
 2    VAV(I) = VAV(I)/QSUM
      WRITE(IOUT,101) (VAV(I),I=1,N)
 101  FORMAT(1H0,T30,30HTHE NORMALIZED INITIAL WEIGHTS  ///(T10,5E20.6))
      DO 3 I=1,N
      IJX = KN + I
 3    X(IJX) = VAV(I)
      RETURN
      END

C
      SUBROUTINE FINAL(VAV,S1,S2,S3,S4,NS,IOUT,RHS,IKX)
      DIMENSION S1(5),S2(5),S3(5),S4(5)
      DIMENSION VAV(5)
      DIMENSION RHS(5,4)
      COMMON ATOTL
      JKX = IKX
      DO 10 I=1,N
 10   VAV(I) = (S1(I) + S2(I) + S3(I) + S4(I))/ ATOTL
      WRITE(IOUT,140) (VAV(I),I=1,N)
 140  FORMAT(1H0,///T10,43HTHE OUTPUT WEIGHTS FOR FLETCHER AND POWELL  /
     1/(T10,5E20.6))
      RETURN
      END
```

```
C
C     ...............................................................
C
C        SUBROUTINE EIGEN
C
C        PURPOSE
C           COMPUTE EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC
C           MATRIX
C
C        USAGE
C           CALL EIGEN(A,R,N,MV)
C
C        DESCRIPTION OF PARAMETERS
C           A - ORIGINAL MATRIX (SYMMETRIC), DESTROYED IN COMPUTATION.
C               RESULTANT EIGENVALUES ARE DEVELOPED IN DIAGONAL OF
C               MATRIX A IN DESCENDING ORDER.
C           R - RESULTANT MATRIX OF EIGENVECTORS (STORED COLUMNWISE,
C               IN SAME SEQUENCE AS EIGENVALUES)
C           N - ORDER OF MATRICES A AND R
C           MV- INPUT CODE
C                   0   COMPUTE EIGENVALUES AND EIGENVECTORS
C                   1   COMPUTE EIGENVALUES ONLY (R NEED NOT BE
C                       DIMENSIONED BUT MUST STILL APPEAR IN CALLING
C                       SEQUENCE)
C
C        REMARKS
C           ORIGINAL MATRIX A MUST BE REAL SYMMETRIC (STORAGE MODE=1)
C           MATRIX A CANNOT BE IN THE SAME LOCATION AS MATRIX R
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           DIAGONALIZATION METHOD ORIGINATED BY JACOBI AND ADAPTED
C           BY VON NEUMANN FOR LARGE COMPUTERS AS FOUND IN 'MATHEMATICAL
C           METHODS FOR DIGITAL COMPUTERS', EDITED BY A. RALSTON AND
C           H.S. WILF, JOHN WILEY AND SONS, NEW YORK, 1962, CHAPTER 7
C
C     ...............................................................
C
      SUBROUTINE EIGEN(A,R,N,MV)
      DIMENSION A(1),R(1)
C
C        ............................................................
C
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C        STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,R,ANORM,ANRMX,THR,X,Y,SINX,SINX2,COSX,
C    1                 COSX2,SINCS,RANGE
C
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
```

```
C        ROUTINE.
C
C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO
C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  SQRT IN STATEMENTS
C        40, 68, 75, AND 78 MUST BE CHANGED TO DSQRT.  ABS IN STATEMENT
C        62 MUST BE CHANGED TO DABS. THE CONSTANT IN STATEMENT 5 SHOULD
C        BE CHANGED TO 1.0D-12.
C
C        ..............................................................
C
C        GENERATE IDENTITY MATRIX
C
    5 RANGE=1.0E-6
      IF(MV-1) 10,25,10
   10 IQ=-N
      DO 20 J=1,N
      IQ=IQ+N
      DO 20 I=1,N
      IJ=IQ+I
      R(IJ)=0.0
      IF(I-J) 20,15,20
   15 R(IJ)=1.0
   20 CONTINUE
C
C        COMPUTE INITIAL AND FINAL NORMS (ANORM AND ANORMX)
C
   25 ANORM=0.0
      DO 35 I=1,N
      DO 35 J=I,N
      IF(I-J) 30,35,30
   30 IA=I+(J*J-J)/2
      ANORM=ANORM+A(IA)*A(IA)
   35 CONTINUE
      IF(ANORM) 165,165,40
   40 ANORM=1.414*SQRT(ANORM)
      ANRMX=ANORM*RANGE/FLOAT(N)
C
C        INITIALIZE INDICATORS AND COMPUTE THRESHOLD, THR
C
      IND=0
      THR=ANORM
   45 THR=THR/FLOAT(N)
   50 L=1
   55 M=L+1
C
C        COMPUTE SIN AND COS
C
   60 MQ=(M*M-M)/2
      LQ=(L*L-L)/2
      LM=L+MQ
   62 IF( ABS(A(LM))-THR) 130,65,65
   65 IND=1
      LL=L+LQ
      MM=M+MQ
```

```
      X=0.5*(A(LL)-A(MM))
   68 Y=-A(LM)/ SQRT(A(LM)*A(LM)+X*X)
      IF(X) 70,75,75
   70 Y=-Y
   75 SINX=Y/ SQRT(2.0*(1.0+( SQRT(1.0-Y*Y))))
      SINX2=SINX*SINX
   78 COSX= SQRT(1.0-SINX2)
      COSX2=COSX*COSX
      SINCS =SINX*COSX
C
C        ROTATE L AND M COLUMNS
C
      ILQ=N*(L-1)
      IMQ=N*(M-1)
      DO 125 I=1,N
      IQ=(I*I-I)/2
      IF(I-L) 80,115,80
   80 IF(I-M) 85,115,90
   85 IM=I+MQ
      GO TO 95
   90 IM=M+IQ
   95 IF(I-L) 100,105,105
  100 IL=I+LQ
      GO TO 110
  105 IL=L+IQ
  110 X=A(IL)*COSX-A(IM)*SINX
      A(IM)=A(IL)*SINX+A(IM)*COSX
      A(IL)=X
  115 IF(MV-1) 120,125,120
  120 ILR=ILQ+I
      IMR=IMQ+I
      X=R(ILR)*COSX-R(IMR)*SINX
      R(IMR)=R(ILR)*SINX+R(IMR)*COSX
      R(ILR)=X
  125 CONTINUE
      X=2.0*A(LM)*SINCS
      Y=A(LL)*COSX2+A(MM)*SINX2-X
      X=A(LL)*SINX2+A(MM)*COSX2+X
      A(LM)=(A(LL)-A(MM))*SINCS+A(LM)*(COSX2-SINX2)
      A(LL)=Y
      A(MM)=X
C
C        TESTS FOR COMPLETION
C
C        TEST FOR M = LAST COLUMN
C
  130 IF(M-N) 135,140,135
  135 M=M+1
      GO TO 60
C
C        TEST FOR L = SECOND FROM LAST COLUMN
C
  140 IF(L-(N-1)) 145,150,145
  145 L=L+1
```

```
      GO TO 55
  150 IF(IND-1) 160,155,160
  155 IND=0
      GO TO 50
C
C        COMPARE THRESHOLD WITH FINAL NORM
C
  160 IF(THR-ANRMX) 165,165,45
C
C        SORT EIGENVALUES AND EIGENVECTORS
C
  165 IQ=-N
      DO 185 I=1,N
      IQ=IQ+N
      LL=I+(I*I-I)/2
      JQ=N*(I-2)
      DO 185 J=I,N
      JQ=JQ+N
      MM=J+(J*J-J)/2
      IF(A(LL)-A(MM)) 170,185,185
  170 X=A(LL)
      A(LL)=A(MM)
      A(MM)=X
      IF(MV-1) 175,185,175
  175 DO 180 K=1,N
      ILR=IQ+K
      IMR=JQ+K
      X=R(ILR)
      R(ILR)=R(IMR)
  180 R(IMR)=X
  185 CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE FILUP(X,N,INPUT)
      DIMENSION X(25),L(6)
      REWIND 10
      READ(INPUT,101) NWTS,NWT1,WT2,WT3
  101 FORMAT(2I5,2E20.6)
      READ(INPUT,102) (L(I),I=1,6)
  102 FORMAT(6I2)
      WRITE(INPUT,103) (X(I),I=1,N)
  103 FORMAT(4E20.6)
      REWIND 10
      RETURN
      END
```

```
C
      SUBROUTINE FINAL(XDOBL,S1,S2,S3,S4,NS,N,IOUT,RHS,IX)
      DIMENSION A(5,5),B(5,5),RHS(5,4),RHD(5,1),EIGN(4,5),XSTAR(5),
     1   XDOBL(5),S1(5),S2(5),S3(5),S4(5)
      DO 55 I=1,5
 55   XDOBL(I) = 0.0E0
      NSETS = NS + 1
      DO 10 JI=1,NS
      J = JI + 1
      DO 10 I=1,JI
 10   A(I,J) = RHS(I,JI)
      DO 11 I=1,NSETS
 11   A(I,I) = 1.E0
      DO 12 J=1,NSETS
      DO 12 I=1,J
 12   A(J,I) = A(I,J)
      DO 13 J=1,NSETS
      JY = J
      IF(IX - J) 14,13,18
 14   JY = JY - 1
 18   DO 16 I=1,J
      IY = I
      IF(IX - I) 15,16,20
 15   IY = IY - 1
 20   B(IY,JY) = A(I,J)
 16   CONTINUE
 13   CONTINUE
      DO 21 J=1,NS
      DO 21 I=1,J
 21   B(J,I) = B(I,J)
      DO 22 I=1,NS
 22   RHD(I,1) = RHS(IX,I)
      WRITE(IOUT,113)
 113  FORMAT(1H0,T10,16HTHE COEFFICIENTS    )
      DO 24 I=1,NS
      WRITE(IOUT,103) (B(I,J),J=1,NS)
 103  FORMAT(1H0,T10,4E20.6)
 24   CONTINUE
      WRITE(IOUT,114)
 114  FORMAT(1H0,T10,19HTHE RIGHT HAND SIDE    )
      WRITE(IOUT,104) (RHD(I,1),I=1,NS)
 104  FORMAT(1H0,T10,4E20.6)
C
C
      CALL MATIL(B,NS,1,DETRM,ID,IOUT,RHD)
C
C
      WRITE(IOUT,105)  IX,(RHD(I,1),I=1,NS)
 105  FORMAT(1H0,T10,4HTHE ,I2,14H SET SOLUTIONS //(T10,4E20.6))
      DO 30 I=1,N
      EIGN(1,I) = S1(I)
      EIGN(2,I) = S2(I)
      EIGN(3,I) = S3(I)
 30   EIGN(4,I) = S4(I)
```

```
      WRITE(IOUT.200)
200   FORMAT(1HO.T10.23HTHE PACKED EIGENVECTOR    )
      DO 40 I=1.NS
40    WRITE(IOUT.201) (EIGN(I.J).J=1.N)
201   FORMAT(1HO.T10.5E20.6)
      DO 31 J=1.NS
31    XSTAR(J) = RHO(J.1)*RHS(IX.J)
      WRITE(IOUT.106) (XSTAR(I).I=1.NS)
106   FORMAT(1HO.T10.15HTHE SOLUTION X*  //(T10.4E20.6))
      DO 32 J=1.N
      DO 32 K=1.NS
32    XDOBL(J) = XDOBL(J) + XSTAR(K)*EIGN(K.J)
      WRITE(IOUT.108) (XDOBL(I).I=1.N)
108   FORMAT(1HO.T10.17HTHE SOLUTIONS X**  //(T10.4E20.6))
      RETURN
      END
```

```
C
      SUBROUTINE MATIL(A.N1.M1.DETRM.IO.IOUT.B)
      DIMENSION A(5.5).B(5.1).INDEX(5.3)
      EQUIVALENCE (IROW.JROW).(ICOLM.JCOLM).(AMAX.T.SWAPP)
      DE = 1.E0
      DZ=0.E0
      M=M1
      N=N1
      DETRM=DE
      DO 20 J=1.N
   20 INDEX(J.3)=DZ
      DO 550 I=1.N
      AMAX=DZ
      DO 105 J=1.N
      IF(INDEX(J.3)-1) 60.105.60
   60 DO 100 K=1.N
      IF(INDEX(K.3)-1) 80.100.715
   80 IF(AMAX-ABS(A(J.K))) 85.100.100
   85 IROW=J
      ICOLM=K
      AMAX=ABS(A(J.K))
  100 CONTINUE
  105 CONTINUE
      INDEX(ICOLM.3)=INDEX(ICOLM.3)+1
      INDEX(I.1)=IROW
      INDEX(I.2)=ICOLM
      IF(IROW-ICOLM) 140.310.140
  140 DETRM=-DETRM
      DO 200 L=1.N
      SWAPP=A(IROW.L)
      A(IROW.L)=A(ICOLM.L)
  200 A(ICOLM.L)=SWAPP
      IF(M) 310.310.210
  210 DO 250 L=1.M
      SWAPP=B(IROW.L)
      B(IROW.L)=B(ICOLM.L)
  250 B(ICOLM.L)=SWAPP
  310 PIVOT=A(ICOLM.ICOLM)
      DETRM=DETRM*PIVOT
      A(ICOLM.ICOLM)=DE
      DO 350 L=1.N
  350 A(ICOLM.L)=A(ICOLM.L)/PIVOT
      IF(M) 380.380.360
  360 DO 370 L=1.M
  370 B(ICOLM.L)=B(ICOLM.L)/PIVOT
  380 DO 550 L1=1.N
      IF(L1-ICOLM) 400.550.400
  400 T=A(L1.ICOLM)
      A(L1.ICOLM)=DZ
      DO 450 L=1.N
  450 A(L1.L)=A(L1.L)-A(ICOLM.L)*T
      IF(M) 550.550.460
  460 DO 500 L=1.M
  500 B(L1.L)=B(L1.L)-B(ICOLM.L)*T
```

```
550 CONTINUE
    DO 710 I=1,N
    L=N+1-I
    IF(INDEX(L,1)-INDEX(L,2)) 630,710,630
630 JROW=INDEX(L,1)
    JCOLM=INDEX(L,2)
    DO 705 K=1,N
    SWAPP=A(K,JROW)
    A(K,JROW)=A(K,JCOLM)
    A(K,JCOLM)=SWAPP
705 CONTINUE
710 CONTINUE
    DO 730 K=1,N
    IF(INDEX(K,3)-1) 715,720,715
715 ID=2
    GO TO 740
720 CONTINUE
730 CONTINUE
    ID=1
740 CONTINUE
    WRITE(IOUT,101)
101 FORMAT(1H0,T10,14HTHE INVERSE IS  /)
    DO 10 I=1,N
10  WRITE(IOUT,102) (A(I,J),J=1,N)
102 FORMAT(1H0,T10,5E20.6)
    WRITE(IOUT,111) DETRM
111 FORMAT(1H0,T10,17HTHE DETERMINANT =  , E20.6)
    RETURN
    END
```

APPENDIX C

COMPUTER PROGRAM FOR FLETCHER AND POWELL METHOD

FOR MINIMIZATION OF A GIVEN FUNCTION

"F P M"

```
      EXTERNAL FUNCT
      DIMENSION H(400).X(25).G(25)
      COMMON IFLAG.RR(5.5.25).NRST(5).NSET
      COMMON S(5)
      DATA NRST.N1.N2.N3.N4.N5/10*0/
      DATA INPUT.IOUT/10.6/
      REWIND 10
C
      CALL IO(N.LIMIT.EST.EPS.X.INPUT.IOUT)
C
C
      CALL FMFP(FUNCT.N.X.F.G.EST.EPS.LIMIT.IER.H.KOUNT)
C
      WRITE(IOUT.113) (X(I).I=1.N)
 113  FORMAT(1HO//.T10.37HFLETCHER POWELL RESULTED VECTOR VALUE ///(T10.
     15E20.6))
      WRITE(IOUT.114) (G(I).I=1.N)
114   FORMAT(1HO/.T10.12HTHE GRADIENT //(T10.5E20.6))
      WRITE(IOUT.115) F
115   FORMAT(1HO/.T10.18HTHE FUNCTION VALUE .E20.6)
      F1 = EXP(F)
      WRITE(IOUT.205) F1
 205  FORMAT(1HO.T20.19HTHE EXP(LOG(DET)) =    .E20.6)
      IER = IER + 2
      GO TO (1.2.3.4).IER
 1    IER = IER - 2
      WRITE(IOUT.116) IER
116   FORMAT(1HO.T10. 5HIER = .I5.31H  ERROR IN GRADIENT CALCULATION )
      GO TO 99
 2    IER = IER - 2
      WRITE(IOUT.117) IER
117   FORMAT(1HO.T10. 5HIER = .I5.28H   CONVERGENCE WAS OBTAINED. )
      GO TO 99
 3    IER = IER - 2
      WRITE(IOUT.118) IER
118   FORMAT(1HO.T10. 5HIER = .I5.37H   NO CONVERGENCE IN LIMIT ITERATIO
     1N )
      GO TO 99
 4    IER = IER - 2
      WRITE(IOUT.119) IER
119   FORMAT(1HO.T10. 5HIER = .I5.26H   NO MINIMUM VALUE EXIST. )
 99   CONTINUE
      NR = 0
      NPJ = 1
      DO 132 I=1.NSET
      NR = NR + NRST(I)
      DO 133 J=NPJ.NR
 133  X(J) = X(J)/S(I)
      NPJ = NR + 1
 132  CONTINUE
      NA = N
      WRITE(IOUT.123)  (X(I).I=1.NA)
 123  FORMAT(1HO.T10.34HTHE NORMALIZED WHOLE WEIGHT-VECTOR  //(T10.5E20.
     16))
```

```
      WRITE(IOUT.201) KOUNT
201   FORMAT(1HO///T10.41HTHE TOTAL NUMBER OF ITERATIONS REQUIRED = .I8)
      REWIND 10
      WRITE(INPUT.101) N.LIMIT.EST.EPS
101   FORMAT(2I5.2E20.6)
      WRITE(INPUT.106) NSET .NRST(I).I=1.5)
106   FORMAT(6I2)
      WRITE(INPUT.804) (X(I).I=1.N)
804   FORMAT(4E20.6)
      REWIND 10
      STOP
      END
      SUBROUTINE IO(N.LIMIT.EST.EPS.X.INPUT.IOUT)
      DIMENSION X(25).DR(5.5).IETZ(6).R(15.5.5)
      COMMON IFLAG.RR(5.5.25).NRST(5).NSET
      READ(INPUT.101) N.LIMIT.EST.EPS
101   FORMAT(2I5.2E20.6)
      READ(INPUT.103) NSET.(NRST(I).I=1.5)
103   FORMAT(6I2)
      KSET = 3
      IF(NSET - 2) 2.1.2
2     KSET = 6
      IF(NSET - 3) 3.1.3
3     KSET = 10
      IF(NSET - 4) 4.1.4
4     KSET = 15
1     CONTINUE
      IETZ(1) = NSET
      DO 6 I=2.6
      J = I - 1
6     IETZ(J) = NRST(I)
      READ(INPUT,102) (X(I),I=1.N)
102   FORMAT(4E20.6)
      DO 3311 I=1.NSET
      JA = NRST(I)
3311  READ(INPUT.801) (NT(I.J).J=1.JA)
801   FORMAT(5I8)
      READ(INPUT.801) NTAL
      DO 3312 JA=1.NSET
3312  READ(INPUT.802) NO.MK.((DR(I.J).J=1.NO).I=1.MK)
802   FORMAT(2I2.(5E20.6))
      WRITE(IOUT.110)
110   FORMAT(1H1.T25.57HFLETCHER AND POWELL METHOD OF MINIMIZATION OF A
     1FUNCTION. )
      WRITE(IOUT.111) N.LIMIT
111   FORMAT(///.T20.23HNUMBER OF VARIBLES: N = .I2,35H      MAXIMUM OF
     1ITERATION: LIMIT = .I4)
      WRITE(IOUT.112) EPS.EST
112   FORMAT(///.T20.24HPERMISSIBLE ERROR: EPS = .E20.6.//T20.46HESTIMAT
     1ED MINIMUM VALUE OF THE FUNCTION: EST = .E20.6)
      WRITE(IOUT.137)
137   FORMAT(1HO.T20.29HTHE ESTIMATED INITIAL WEIGHTS    )
      DO 136 I=1.N
136   WRITE(IOUT.135) I.X(I)
```

```
135  FORMAT(1H0.T25.2HX(,I2.4H)  =.E20.6)
     WRITE(IOUT.224)
224  FORMAT(1H0.T35.16HTHE INPUT MATRIX    )
     WRITE(IOUT.250)
250  FORMAT(1H0//T10.3HITH.2X.3HJTH.2X.3HROW/T10.3HSET.2X.3HSET.2X.3HNO
    1..7X.8HELEMENTS  )
     K = 1
     DO 10 J=1.NSET
     JSET = NRST(J)
     DO 10 I=1.J
     ISET = NRST(I)
     READ(INPUT.145) ((R(K.IX.JX).JX=1.JSET).IX=1.ISET)
145  FORMAT(4E20.6)
     DO 16 IXX=1,ISET
     DO 16 JXX=1.JSET
16   DR(IXX.JXX) = R(K.IXX.JXX)
     CALL CONEC(K.IETZ.DR.IOUT)
     K = K + 1
     IF(K - KSET) 10.10.11
10   CONTINUE
11   CONTINUE
     RETURN
     END
```

```
C
      SUBROUTINE CONEC(K.IETZ.DR.IOUT)
      DIMENSION IETZ(6).DR(5.5)
      COMMON IFLAG.RR(5.5.25)
      L = K
      NCOL = 1
      NV = IETZ(1)
      DO 1 J=2.NV
      IF(L - 1) 3. 3. 33
 33   CONTINUE
      L = L - J
      NCOL = NCOL + 1
 1    CONTINUE
 3    NRO = K - (NCOL*(NCOL - 1))/2
      NCDS = IETZ(NRO + 1)
      DO 5 KZ=1.NCDS
      NENT = IETZ(NCOL + 1)
      DO 128 JZ=1.NENT
      IKJ = (JZ - 1)*NCDS + KZ
      RR(NRO.NCOL.IKJ) = DR(KZ.JZ)
      JKI = (KZ - 1)*NENT + JZ
 128  RR(NCOL.NRO.JKI) = DR(KZ.JZ)
      WRITE(IOUT.250) NRO.NCOL.KZ.(DR(KZ.JK).JK=1.NENT)
 250  FORMAT(1H0.T10.I2.3X.I2.3X.I2.3X.4E20.6)
 5    CONTINUE
      RETURN
      END


      FUNCTION AMAX1(X.Y.Z)
      AMAX1 = X
      IF(AMAX1 - Y) 11.12.12
   11  AMAX1 = Y
   12  IF(AMAX1 - Z) 13.14.14
   13 AMAX1 = Z
14    RETURN
      END
```

```
C
C        ...............................................................
C
C           SUBROUTINE MINV
C
C           PURPOSE
C              INVERT A MATRIX
C
C           USAGE
C              CALL MINV(A,N,D,L,M)
C
C           DESCRIPTION OF PARAMETERS
C              A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY
C                  RESULTANT INVERSE.
C              N - ORDER OF MATRIX A
C              D - RESULTANT DETERMINANT
C              L - WORK VECTOR OF LENGTH N
C              M - WORK VECTOR OF LENGTH N
C
C           REMARKS
C              MATRIX A MUST BE A GENERAL MATRIX
C
C           SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C              NONE
C
C           METHOD
C              THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT
C              IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT
C              THE MATRIX IS SINGULAR.
C
C        ...............................................................
C
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
C
C           ............................................................
C
C           IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C           C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C           STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
C           THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C           APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
C           ROUTINE.
C
C           THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO
C           CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT
C           10 MUST BE CHANGED TO DABS.
C
C           ............................................................
C
C           SEARCH FOR LARGEST ELEMENT
```

```
C
      D=1.0
      NK=-N
      DO 80 K=1,N
      NK=NK+N
      L(K)=K
      M(K)=K
      KK=NK+K
      BIGA=A(KK)
      DO 20 J=K,N
      IZ=N*(J-1)
      DO 20 I=K,N
      IJ=IZ+I
   10 IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20
   15 BIGA=A(IJ)
      L(K)=I
      M(K)=J
   20 CONTINUE
C
C        INTERCHANGE ROWS
C
      J=L(K)
      IF(J-K) 35,35,25
   25 KI=K-N
      DO 30 I=1,N
      KI=KI+N
      HOLD=-A(KI)
      JI=KI-K+J
      A(KI)=A(JI)
   30 A(JI) =HOLD
C
C        INTERCHANGE COLUMNS
C
   35 I=M(K)
      IF(I-K) 45,45,38
   38 JP=N*(I-1)
      DO 40 J=1,N
      JK=NK+J
      JI=JP+J
      HOLD=-A(JK)
      A(JK)=A(JI)
   40 A(JI) =HOLD
C
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS
C        CONTAINED IN BIGA)
C
   45 IF(BIGA) 48,46,48
   46 D=0.0
      RETURN
   48 DO 55 I=1,N
      IF(I-K) 50,55,50
   50 IK=NK+I
      A(IK)=A(IK)/(-BIGA)
   55 CONTINUE
```

```
C
C        REDUCE MATRIX
C
      DO 65 I=1,N
      IK=NK+I
      HOLD=A(IK)
      IJ=I-N
      DO 65 J=1,N
      IJ=IJ+N
      IF(I-K) 60,65,60
   60 IF(J-K) 62,65,62
   62 KJ=IJ-I+K
      A(IJ)=HOLD*A(KJ)+A(IJ)
   65 CONTINUE
C
C        DIVIDE ROW BY PIVOT
C
      KJ=K-N
      DO 75 J=1,N
      KJ=KJ+N
      IF(J-K) 70,75,70
   70 A(KJ)=A(KJ)/BIGA
   75 CONTINUE
C
C        PRODUCT OF PIVOTS
C
      D=D*BIGA
C
C        REPLACE PIVOT BY RECIPROCAL
C
      A(KK)=1.0/BIGA
   80 CONTINUE
C
C        FINAL ROW AND COLUMN INTERCHANGE
C
      K=N
  100 K=(K-1)
      IF(K) 150,150,105
  105 I=L(K)
      IF(I-K) 120,120,108
  108 JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=-A(JI)
  110 A(JI) =HOLD
  120 J=M(K)
      IF(J-K) 100,100,125
  125 KI=K-N
      DO 130 I=1,N
      KI=KI+N
      HOLD=A(KI)
```

```
      JI=KI-K+J
      A(KI)=-A(JI)
  130 A(JI) =HOLD
      GO TO 100
  150 RETURN
      END
```

```
C     THE SUBROUTINE FMFP
      SUBROUTINE FMFP(FUNCT.N.X.F.G.EST.EPS.LIMIT.IER.H.KOUNT)
      DIMENSION H(1).X(1).G(1)
      COMMON IFLAG
C
C     FUNCT-USER WRITEN SUBROUTINE CONCERNING THE FUNCTION TO BE
C           MINIMIZING IT MUST BE OF THE FORM
C                 SUBROUTINE FUNCT(N.ARG.VAL.GRAD)
C           AND MUST SERVE THE FOLLOWING PURPOSE.
C           FOR EACH N-DIMENSIONAL ARGUMENT VECTOR ARG.
C           FUNCTION VALUE AND GRADIENT VECTOR MUST BE COMPUTED AND.ON
C           RETURN. STORE IN VAL AND GRAD RESPECTIVELY.
C     N    -NUMBER OF VARIABLES.
C     X    -VECTOR OF DIMENSION N CONTAINING THE INITIAL ARGUMENT. WHERE
C           THE INTERATION STARTS. ON RETURN X HOLDS THE ARGUMENT
C           CORRESPONDING TO THE COMPUTED MINIMUM FUNCTION VALUE.
C     G    -VECTOR OF DIMENSION N CONTAINING THE GRADIENT VECTOR
C           CORRESPONDING TO THE MINIMU  ON RETURN I.E. G = G(X).
C     EST  -IS AN ESTIMATE OF THE MINIMUM FUNCTION VALUE.
C     ESP  -TEST VALUE REPRESENTING THE EXPECTED ABSLUTE ERROR. A
C           REASONABLE CHOICE IS 10**(-6). I.E. SOME WHAT GREATER THAN
C           10**(-D) WHERE D IS THE NUMBER OF SIGNIFICANT DIGITS IN
C           FLOATING POINT REPRESENTATION.
C     LIMIT-MAXIMUM NUMBER OF ITERATION.
C     IER  -ERROR PARAMETERS.
C           IER = 0. CONVERGENCE WAS OBTAINED.
C           IER = 1. NO CONVERGENCE IN LIMIT ITERATION.
C           IER = -1. ERROR IN GRADIENT CALCULATION.
C           IER = 2.LINEAR SEARCH TECHNIQUE INDICATE IT IS A LIKELY THAT
C                  THERE EXISTS NO MINUMUM.
C     H    -WORKING STORAGE OF DIMENSION N*(N + 7)/2.
C
C     REMARKS:
C     1. THE SUBROUTINE NAME REPLACING THE DUMMY ARGUMENT FUNCT. MUST BE
C       DECLARED AS ESTERNAL IN THE CALLING PROGRAM.
C     2. IER IS SET TO 2 IF STEPPING IN ONE OF THE COMPUTED DIRECTIONS.
C        THE FUNCTION WILL NEVER INCREASE WITHIN A TOLERABLE RANGE OF
C        ARGUMENTS.
C        IER=2. MAY OCCUR ALSO IF THE INTERVAL WHERE F INCREASES IS SMAL
C        AND THE INITIAL ARGUMENT WAS RELATIVELY FAR AWAY FROM THE MINIM
C        SUCH THAT THE MINIMUM WAS OVERLEAPED. THIS IS DUE TO THE
C        SEARCH RECHNIQUE WHICH DOUBLE THE STEPSIZE UNTIL A POINT IS FOU
C        WHERE THE FUNCTION INCREASES.
C
C     METHOD: THE METHOD IS DISCRIBED IN THE FOLLOWING ARTICLE.
C             R.FLETCHER AND M.J.D.POWELL "A REPID DESCENT METHOD FOR
C             MINIMIZATION.
C     COMPUTER JOURNAL VOLUME 6. ISS.II.1963. PP. 163-168
C
      IFLAG = 1
      CALL FUNCT(N.X.F.G)
      IFLAG = 0
      IER = 0
      KOUNT = 0
```

```
      N2 = N + N
      N3 = N2 + N
      N31 = N3 + 1
1     K = N31
      DO 4 J=1,N
      H(K) = 1.
      NJ = N - J
      IF(NJ) 5,5,2
2     DO 3 L=1,NJ
      KL = K + L
3     H(KL) = 0.
4     K = KL + 1
5     KOUNT = KOUNT + 1
      OLDF = F
      DO 9 J=1,N
      K = N + J
      H(K) = G(J)
      K = K + N
      H(K) = X(J)
      K = J + N3
      T = 0.
      DO 8 L =1,N
      T = T - G(L)*H(K)
      IF(L-J) 6,7,7
6     K = K + N - L
      GO TO 8
7     K = K + 1
8     CONTINUE
9     H(J) = T
      DY = 0.
      HNRM = 0.
      GNRM = 0.
      DO 10 J=1,N
      HNRM = HNRM + ABS(H(J))
      GNRM = GNRM + ABS(G(J))
10    DY = DY + H(J)*G(J)
      IF(DY) 11,51,51
11    IF(HNRM/GNRM - EPS) 51,51,12
12    FY = F
      ALFA = 2.00*(EST - F)/DY
      AMBDA = 1.
      IF(ALFA) 15,15,13
13    IF(ALFA - AMBDA) 14,15,15
14    AMBDA = ALFA
15    ALFA = 0.
16    FX = FY
      DX = DY
      DO 17 I=1,N
17    X(I) = X(I) + AMBDA*H(I)
      CALL FUNCT(N,X,F,G)
      FY = F
      DY = 0.
      DO 18 I=1,N
18    DY = DY + G(I)*H(I)
```

```
      IF(DY) 19,36,22
19    IF(FY - FX) 20,22,22
20    AMBDA = AMBDA + ALFA
      ALFA = AMBDA
      IF(HNRM*AMBDA - 1.E10) 16,16,21
21    IER = 2
      RETURN
22    T = 0.
23    IF(AMBDA) 24,36,24
24    Z = 3.00*(FX - FY)/AMBDA + DX + DY
      ALFA = AMAX1( ABS(Z), ABS(DX), ABS(DY))
      DALFA = Z/ALFA
      DALFA = DALFA*DALFA - DX/ALFA*DY/ALFA
      IF(DALFA) 51,25,25
25    W = ALFA* SQRT(DALFA)
      ALFA = DY - DX + W + W
      IF(ALFA) 250,251,250
 250  ALFA = (DY - Z + W)/ALFA
      GO TO 252
 251  ALFA = (Z + DY - W)/(Z + DX + Z + DY)
 252   ALFA = ALFA*AMBDA
      DO 26 I=1,N
26    X(I) = X(I) + (T -ALFA)*H(I)
      IFLAG = 1
      CALL FUNCT(N,X,F,G)
      IFLAG = 0
      IF(F - FX) 27,27,28
27    IF(F - FY) 36,36,28
28    DALFA = 0.
      DO 29 I=1,N
29    DALFA = DALFA + G(I)*H(I)
      IF(DALFA) 30,33,33
30    IF(F - FX) 32,31,33
31    IF(DX - DALFA) 32,36,32
32    FX = F
      DX = DALFA
      T = ALFA
      AMBDA = ALFA
      GO TO 23
33    IF(FY - F) 35,34,35
34    IF(DY - DALFA) 35,36,35
35    FY = F
      DY = DALFA
      AMBDA = AMBDA - ALFA
      GO TO 22
36    DO 37 J=1,N
      K = N + J
      H(K) = G(J) - H(K)
      K = N + K
37    H(K) = X(J) -H(K)
      IF(OLDF - F + EPS) 51,38,38
38    IER = 0
      IF(KOUNT - N) 42,39,39
39    T = 0.
```

```
      Z = 0.
      DO 40 J=1,N
      K = N + J
      W = H(K)
      K = K + N
      T = T + ABS(H(K))
40    Z = Z + W*H(K)
      IF(HNRM - EPS) 41,41,42
41    IF(T - EPS) 56,56,42
42    IF(KOUNT - LIMIT) 43,50,50
43    ALFA = 0.
      DO 47 J=1,N
      K = J + N3
      W = 0.
      DO 46 L=1,N
      KL = N + L
      W = W + H(KL)*H(K)
      IF(L - J) 44,45,45
44    K = K + N - L
      GO TO 46
45    K = K + 1
46    CONTINUE
      K = N + J
      ALFA = ALFA + W*H(K)
47    H(J) = W
      IF(Z*ALFA) 48,1,48
48    K = N31
      DO 49 L=1,N
      KL = N2 + L
      DO 49 J=L,N
      NJ = N2 + J
      H(K) = H(K) + H(KL)*H(NJ)/Z - H(L)*H(J)/ALFA
49    K = K + 1
      GO TO 5
50    IER = 1
      RETURN
51    DO 52 J=1,N
      K = N2 + J
52    X(J) = H(K)
      IFLAG = 1
      CALL FUNCT(N,X,F,G)
      IFLAG = 0
      IF(GNRM - EPS) 55,55,53
53    IF(IER) 56,54,54
54    IER = -1
      GO TO 1
55    IER =0
56    RETURN
      END
```

```
C
C     SUBROUTINE FUNCT(N.X.Y.G)
      SUBROUTINE  FUNCT(N.XX.Y.G)
      DIMENSION XX(1).G(1).RHO(5.5).A(25).LLX(5).MMX(5).X(5)
      COMMON IFLAG.RR(5.5.25).NRST(5).NSET
      COMMON S(5)
      KP = 0
      KM = 0
      DO 1 I = 1.NSET
      S(I) = 0.
      NA = NRST(I)
      DO 2 J = 1.NA
      JM = J + KM
 2    S(I) = S(I) + XX(JM)**2
      S(I) = SQRT(S(I))
 1    KM = KM + NA
      NR = 0
      DO 1101 I=1.NSET
1101  NR = NR + NRST(I)
      IF (IFLAG.EQ.1)
     1WRITE(6.81) (XX(JZ).JZ=1.NR)
      IF(IFLAG.EQ.1) WRITE(6.81) (S(JZ).JZ=1.NSET)
81    FORMAT(1H0.T10. 4HXX = .5E20.6)
      IM = 0
      DO 11 I = 1.NSET
      JM = IM
      NA = NRST(I)
      DO 12 J=I.NSET
      NB = NRST(J)
      RHO(I.J) = 0.E0
      DO 13 K = 1.NA
      KA = K + IM
      DO 13 L = 1. NB
      KB = L + JM
      KLX = (L-1)*NA + K
   13 RHO(I.J) = RHO(I.J) + XX(KA)*RR(I.J.KLX)*XX(KB)/S(I)/S(J)
      IF (IFLAG.EQ.1)
     1WRITE(6.82) I.J.RHO(I.J)
82    FORMAT(1H0.T10. 4HRHO( .I2,2H. .I2,3H)= ,E15.6)
   12 JM = JM + NB
   11 IM = IM + NA
      DO 21 I = 1.NSET
      DO 21 J = I.NSET
   21 RHO(J.I) = RHO(I.J)
      DO 22 I = 1.NSET
      DO 22 J=1.NSET
      IJX = (J-1)*NSET + I
   22 A(IJX) = RHO(I.J)
      NN = NSET*NSET
      IF (IFLAG.EQ.1)
     1WRITE(6.83) (A(KX).KX=1.NN)
83    FORMAT(1H0.T10.10HA PACKED =.5E15.6/(T20.4E15.6))
      CALL MINV(A.NSET.DET.LLX.MMX)
      IF (IFLAG.EQ.1)
```

```
      1WRITE(6,83) (A(KX),KX=1,NN)
       Y = ALOG(DET)
       IF (IFLAG.EQ.1)
      1WRITE(6,88) Y
 88    FORMAT(1H0,T10,24HTHE EVALUATED FUNCTION = ,E20.6)
       Y1 = EXP(Y)
       IF(IFLAG.EQ.1)
      1WRITE(6,89) Y1
 89    FORMAT(1H0,T10,19HTHE EXP(LOG(DET)) =,E20.6)
       IM = 0
       IL = 0
       DO 30 I = 1,NSET
       NA = NRST(I)
       KU = NA + IM
       NAM = NA
       DO 31 K = 1,NAM
       KQ = K + IL
       G(KQ) = 0.E0
       KR = K + IM
       JM = 0
       DO 32 J = 1,NSET
       NB = NRST(J)
   38  IXJ = (J-1)*NSET + I
       X(J) = 0.D0
       DO 34 KK = 1,NB
       KKM = KK + JM
       KXKK = (KK-1)*NA + K
       X(J) = X(J) + A(IXJ)*RR(I,J,KXKK)*XX(KKM)/S(J)
  34   CONTINUE
       G(KQ) = G(KQ) + X(J)
   32  JM = JM + NB
       G(KQ) = G(KQ) - XX(KQ)/S(I)
       G(KQ) = 2.D0*G(KQ)/S(I)
   31  CONTINUE
       IM = IM + NA
  30   IL = IL + NA
       IF(IFLAG.EQ.1)  WRITE(6,118) (G(I),I=1,N)
 118   FORMAT(1H0,19X,15HTHE GRADIENT IS /(15X,5E20.6))
       RETURN
       END
```

APPENDIX D

COMPUTER PROGRAM FOR CANONICAL-PARTIAL AND CANONICAL-MULTIPLE CORRELATION MATRIX "CPCM"

```
C
C
C***********************************************************************
C*
C*    THIS IS THE MAIN CALLING PROGRAM.
C*
C***********************************************************************
C
      DIMENSION RST11(10.10).RST12(10.20).RST22(20.20).RESUT(10.10)
      DIMENSION TX(10.10).TY(10.10).R(30).EIGN(30).ATINV(20.20)
     1 .NT(5.5).NRST(5)
      COMMON N1.N2.N3.N4.N5.DZ.DE.IOUT.ML.NSETS
      COMMON X(25)
      COMMON R11(5.5).R12(5.5).R13(5.5).R14(5.5).R15(5.5).R21(5.5).R22(5
     1.5).R23(5.5).R24(5.5).R25(5.5).R31(5.5).R32(5.5).R33(5.5).R34(5.5)
     2.R35(5.5)
      COMMON R41(5.5).R42(5.5).R43(5.5).R44(5.5).R45(5.5).R51(5.5).R52(5
     1.5).R53(5.5).R54(5.5).R55(5.5).PRHO(5.5)
      REWIND 10
      INPUT = 10
      IOUT = 6
      DZ = 0.E0
      DE = 1.E0
      READ(INPUT.301) ML.LIMIT.EST.EPS
 301  FORMAT(2I5.2E20.6)
      READ(INPUT.100) NSETS.(NRST(I).I=1.5)
 100  FORMAT(6I2)
      N1 = NRST(1)
      N2 = NRST(2)
      N3 = NRST(3)
      N4 = NRST(4)
      N5 = NRST(5)
      READ(INPUT.101) (X(I).I=1.ML)
      DO 811 J=1.NSETS
      NA = NRST(J)
      READ(INPUT.802) (NT(J.I).I=1.NA)
 802  FORMAT(5I8)
 811  CONTINUE
      READ(INPUT.802) NTAL
      DO 812 JA=1.NSETS
 812  READ(INPUT.813) ND.MK.((ATINV(I.J).J=1.ND).I=1.MK)
 813  FORMAT(2I2.(5E20.6))
      IF(NSETS-1) 2999.2999.2998
 2998 IF(NSETS-6) 2997.2999.2999
 2999 WRITE(IOUT.2996)
 2996 FORMAT(1H1.T10.81HTHE NUMBER OF SETS OF VARIABLES IS WRONG. CHECK
     1IT. AND REENTER THE WHOLE PROGRAM )
      PAUSE 1110
      GO TO 99
 2997 CONTINUE
      WRITE(IOUT.105)
 105  FORMAT(1H1.T25.46HTHE CANONICAL-PARTIAL AND MULTIPLE CORRELATION /
     1//T35.14HTHE INPUT DATA///// )
      WRITE(IOUT.608) NSETS.N1.N2.N3.N4.N5
```

```
609  FORMAT(1H0,T10,28HTHE NUMBER OF SETS (NSETS) = ,I6//T10,37HTHE NUM
    1BER OF ROWS OF (1,1) SET N1  = ,I6//T10,37HTHE NUMBER OF ROWS OF (
    22,2) SET N2  = ,I6//T10,37HTHE NUMBER OF ROWS OF (3,3) SET N3  = ,
    3I6//T10,37HTHE NUMBER OF ROWS OF (4,4) SET N4  = ,I6//T10,37HTHE N
    4UMBER OF ROWS OF (5,5) SET N5  = ,I6)
     WRITE(IOUT,4202) ML
4202 FORMAT(1H0/T10,24HTHE NUMBER OF WEIGHTS  = ,I6//T10,14HTHE R - MAT
    1RIX  )
     READ(INPUT,101) ((R11(I,J),J=1,N1),I=1,N1)
     READ(INPUT,101) ((R12(I,J),J=1,N2),I=1,N1)
     READ(INPUT,101) ((R22(I,J),J=1,N2),I=1,N2)
     WRITE(IOUT,106) ((R11(I,J),J=1,N1),I=1,N1)
     WRITE(IOUT,106) ((R12(I,J),J=1,N2),I=1,N1)
     WRITE(IOUT,106) ((R22(I,J),J=1,N2),I=1,N2)
     IF(NSETS-2) 3001,3001,3002
3002 CONTINUE
     READ(INPUT,101) ((R13(I,J),J=1,N3),I=1,N1)
     READ(INPUT,101) ((R23(I,J),J=1,N3),I=1,N2)
     READ(INPUT,101) ((R33(I,J),J=1,N3),I=1,N3)
     WRITE(IOUT,106) ((R13(I,J),J=1,N3),I=1,N1)
     WRITE(IOUT,106) ((R23(I,J),J=1,N3),I=1,N2)
     WRITE(IOUT,106) ((R33(I,J),J=1,N3),I=1,N3)
     IF(NSETS-3) 3003,3001,3003
3003 CONTINUE
     READ(INPUT,101) ((R14(I,J),J=1,N4),I=1,N1)
     READ(INPUT,101) ((R24(I,J),J=1,N4),I=1,N2)
     READ(INPUT,101) ((R34(I,J),J=1,N4),I=1,N3)
     READ(INPUT,101) ((R44(I,J),J=1,N4),I=1,N4)
     WRITE(IOUT,106) ((R14(I,J),J=1,N4),I=1,N1)
     WRITE(IOUT,106) ((R24(I,J),J=1,N4),I=1,N2)
     WRITE(IOUT,106) ((R34(I,J),J=1,N4),I=1,N3)
     WRITE(IOUT,106) ((R44(I,J),J=1,N4),I=1,N4)
     IF(NSETS-4) 3004,3001,3004
3004 CONTINUE
     READ(INPUT,101) ((R15(I,J),J=1,N5),I=1,N1)
     READ(INPUT,101) ((R25(I,J),J=1,N5),I=1,N2)
     READ(INPUT,101) ((R35(I,J),J=1,N5),I=1,N3)
     READ(INPUT,101) ((R45(I,J),J=1,N5),I=1,N4)
     READ(INPUT,101) ((R55(I,J),J=1,N5),I=1,N5)
     WRITE(IOUT,106) ((R15(I,J),J=1,N5),I=1,N1)
     WRITE(IOUT,106) ((R25(I,J),J=1,N5),I=1,N2)
     WRITE(IOUT,106) ((R35(I,J),J=1,N5),I=1,N3)
     WRITE(IOUT,106) ((R45(I,J),J=1,N5),I=1,N4)
     WRITE(IOUT,106) ((R55(I,J),J=1,N5),I=1,N5)
3001 CONTINUE
 101 FORMAT(4E20.6)
 106 FORMAT(1H0,//(T10,5E20.6))
     WRITE(IOUT,134)
134  FORMAT(1H0,T10,56HTHE WEIGHTS FROM FLETCHER AND POWELL MINIMIZING
    1 PROGRAM  )
     DO 136 I=1,ML
136  WRITE(IOUT,135) I,X(I)
135  FORMAT(1H0,T15,2HX(,I2,4H)  = ,E20.6)
     CALL SWAP1(R12,R13,R14,R15,R21,R23,R24,R25,R31,R32,R34,R35,R41,R42
```

```
     1.R43.R45.R51.R52.R53.R54)
      KL=1
      IM=1
      IN=2
      N=N1
      M=N2
      L=N3
      KN=N4
      KM=N5
      CALL RSTAR(R11.R12.R13.R14.R15.R22.R23.R24.R25.R33.R34.R35.R44.R45
     1.R55.RST11.RST12.RST22.N.M.L.KN.KM.NSETS.IOUT)
   10 CONTINUE
      CALL MATIV(RST22.M.O.DETRM.IO.IOUT)
      NN = 1
      CALL TRILW(RST11.N.TX.TY.MK.NN.DZ.ATINV.IOUT)
      CALL MULT2(TX.RST22.RST12.N.MK.M.RESUT.DZ.IOUT.KL)
      DO 60 I=1.MK
      DO 60 J=I.MK
      IJ=(J*(J-1))/2 + I
   60 EIGN(IJ)=RESUT(I.J)
      CALL EIGEN(EIGN.R.MK.0)
      WRITE(IOUT.102) KL.EIGN(1)
  102 FORMAT(1HO.T10.3HTHE.I3.12H EIGEN-VALUE //(T10.5E20.6))
      WRITE(IOUT.120) (R(I).I=1.MK)
  120 FORMAT(1HO.T10.27HTHE ASSOCIATED EIGEN-VECTOR //(T10.5E20.6))
      RHO= SQRT( ABS(EIGN(1)))
      WRITE(IOUT.111) IM.IN
  111 FORMAT(1HO.T10.38HTHE CANONICAL-PARTIAL CORRELATION. SET.I2.7H VS
     1SET.I2.18H GIVEN THE OTHERS )
      PRHO(IM.IN)=RHO
      WRITE(IOUT.103) PRHO(IM.IN)
  103 FORMAT(1HO.T20.E20.6)
      IF(NSETS - 2) 4000.11.4000
 4000 CONTINUE
      GO TO (1.2.3.4.5.6.7.8.9.11).KL
    1 CONTINUE
      KL=KL + 1
      IM = 1
      IN=3
      N = N1
      M = N3
      L = N2
      KN = N4
      KM = N5
      CALL RSTAR(R11.R13.R12.R14.R15.R33.R32.R34.R35.R22.R24.R25.R44.R45
     1.R55.RST11.RST12.RST22.N.M.L.KN.KM.NSETS.IOUT)
      GO TO 10
    2 CONTINUE
      IM = 2
      IN = 3
      N = N2
      M = N3
      L = N1
      KN = N4
```

```
      KM = N5
      CALL RSTAR(R22,R23,R21,R24,R25,R33,R31,R34,R35,R11,R14,R15,R44,R45
     1,R55,RST11,RST12,RST22,N,M,L,KN,KM,NSETS,IOUT)
      GO TO 10
    3 CONTINUE
      IF(NSETS-3) 4004,11,4004
 4004 CONTINUE
      KL = KL + 1
      IM = 1
      IN = 4
      N = N1
      M = N4
      L = N2
      KN = N3
      KM = N5
      CALL RSTAR(R11,R14,R12,R13,R15,R44,R42,R43,R45,R22,R23,R25,R33,R35
     1,R55,RST11,RST12,RST22,N,M,L,KN,KM,NSETS,IOUT)
      GO TO 10
    4 CONTINUE
      KL = KL + 1
      IM = 2
      IN = 4
      N = N2
      M = N4
      L = N1
      KN = N3
      KM = N5
      CALL RSTAR(R22,R24,R21,R23,R25,R44,R41,R43,R45,R11,R13,R15,R33,R35
     1,R55,RST11,RST12,RST22,N,M,L,KN,KM,NSETS,IOUT)
      GO TO 10
    5 CONTINUE
      KL = KL + 1
      IM = 3
      IN = 4
      N = N3
      M = N4
      L = N1
      KN = N2
      KM = N5
      CALL RSTAR(R33,R34,R31,R32,R35,R44,R41,R42,R45,R11,R12,R15,R22,R25
     1,R55,RST11,RST12,RST22,N,M,L,KN,KM,NSETS,IOUT)
      GO TO 10
    6 CONTINUE
      IF(NSETS-4) 4006,11,4006
 4006 CONTINUE
      KL= KL + 1
      IM = 1
      IN = 5
      N = N1
      M = N5
      L = N2
      KN = N3
      KM = N4
      CALL RSTAR(R11,R15,R12,R13,R14,R55,R52,R53,R54,R22,R23,R24,R33,R34
```

```
     1.R44.RST11.RST12.RST22.N.M.L.KN.KM.NSETS.IOUT)
      GO TO 10
    7 CONTINUE
      KL = KL + 1
      IM = 2
      IN = 5
      N = N2
      M = N5
      L = N1
      KN = N3
      KM = N4
      CALL RSTAR(R22.R25.R21.R23.R24.R55.R51.R53.R54.R11.R13.R14.R33.R34
     1.R44.RST11.RST12.RST22.N.M.L.KN.KM.NSETS.IOUT)
      GO TO 10
    8 CONTINUE
      KL = KL + 1
      IM = 3
      IN = 5
      N = N3
      M = N5
      L = N1
      KN = N2
      KM = N4
      CALL RSTAR(R33.R35.R31.R32.R34.R55.R51.R52.R54.R11.R12.R14.R22.R24
     1.R44.RST11.RST12.RST22.N.M.L.KN.KM.NSETS.IOUT)
      GO TO 10
    9 CONTINUE
      KL = KL + 1
      IM = 4
      IN = 5
      N = N4
      M = N5
      L = N1
      KN = N2
      KM = N3
      CALL RSTAR(R44.R45.R41.R42.R43.R55.R51.R52.R53.R11.R12.R13.R22.R23
     1.R33.RST11.RST12.RST22.N.M.L.KN.KM.NSETS.IOUT)
      GO TO 10
   11 CONTINUE
      KL = KL + 1
      CALL PARHO(PRHO.IN.IOUT)
      WRITE(IOUT.1111)
 1111 FORMAT(1H1)
 1131 CALL MULCR
 99   STOP
      END
```

```
C
      SUBROUTINE SWAP1(RT12,RT13.RT14.RT15.RT21,RT23.RT24.RT25.RT31.RT32
     1.RT34.RT35.RT41.RT42,RT43.RT45.RT51.RT52,RT53.RT54)
      DIMENSION RT12(5.5).RT13(5.5).RT14(5.5).RT15(5.5).RT21(5.5).RT23(5
     1.5).RT24(5.5).RT25(5.5).RT31(5.5).RT32(5.5).RT34(5.5).RT35(5.5).RT
     241(5.5).RT42(5.5).RT43(5.5).RT45(5.5).RT51(5.5).RT52(5.5).RT53(5.5
     3).RT54(5.5)
      COMMON N1.N2.N3.N4.N5.DZ.DE.IOUT.ML.NSETS
      DO 10 I=1.N1
      DO 10 J=1.N2
   10 RT21(J.I)=RT12(I.J)
      IF(NSETS - 2) 200.99.200
  200 CONTINUE
      DO 11 I=1.N1
      DO 11 J=1.N3
   11 RT31(J.I)=RT13(I.J)
      DO 12 I=1.N2
      DO 12 J=1.N3
   12 RT32(J.I)=RT23(I.J)
      IF(NSETS-3) 210.99.210
  210 CONTINUE
      DO 13 I=1.N1
      DO 13 J=1.N4
   13 RT41(J.I)=RT14(I.J)
      DO 14 I=1.N2
      DO 14 J=1.N4
   14 RT42(J.I)=RT24(I.J)
      DO 15 I=1.N3
      DO 15 J=1.N4
   15 RT43(J.I)=RT34(I.J)
      IF(NSETS-4) 211.99.211
  211 CONTINUE
      DO 16 I=1.N1
      DO 16 J=1.N5
   16 RT51(J.I)=RT15(I.J)
      DO 17 I=1.N2
      DO 17 J=1.N5
   17 RT52(J.I)=RT25(I.J)
      DO 18 I=1.N3
      DO 18 J=1.N5
   18 RT53(J.I)=RT35(I.J)
      DO 19 I=1.N4
      DO 19 J=1.N5
   19 RT54(J.I)=RT45(I.J)
   99 RETURN
      END
```

```
C
      SUBROUTINE RSTAR(RS11.RS12.RS13.RS14.RS15.RS22.RS23.RS24.RS25.RS33
     1.RS34.RS35.RS44.RS45.RS55.RST11.RST12.RST22.NN.NM.NL.NKN.NKM.NSETS
     2.IOUT)
      DIMENSION RS11(5.5).RS12(5.5).RS13(5.5).RS14(5.5).RS15(5.5).RS22(5
     1.5).RS23(5.5).RS24(5.5).RS25(5.5).RS33(5.5).RS34(5.5).RS35(5.5).RS
     244(5.5).RS45(5.5).RS55(5.5)
      DIMENSION WORK1(5.5).WORK2(5.5).WORK3(5.5)
      DIMENSION WORK4(15.15).WORK5(15.15).WORK6(15.15)
      DIMENSION RST11(10.10).RST12(10.20).RST22(20.20)
      DIMENSION PAK33(20.20)
      DIMENSION PAK13(5.15).PAK23(5.15)
      DATA WORK1.WORK2.WORK3.PAK13.PAK23/225*0.E0/
      DATA WORK4.WORK5.WORK6/675*0.E0/
      DATA PAK33/400*0.E0/
      IF(NSETS - 2) 300.96.300
 300  CONTINUE
      DO 30 I=1.NN
      DO 30 J=1.NL
      PAK13(I.J)=RS13(I.J)
 30   CONTINUE
      DO 31 I=1.NM
      DO 31 J=1.NL
      PAK23(I.J)=RS23(I.J)
 31   CONTINUE
      DO 32 I=1.NL
      DO 32 J=1.NL
   32 PAK33(I.J)=RS33(I.J)
      IF(NSETS-3) 301.96.301
  301 CONTINUE
      DO 33 I=1.NN
      DO 33 J=1.NKN
      INKN = NL + J
      PAK13(I.INKN)=RS14(I.J)
 33   CONTINUE
      DO 34 I=1.NM
      DO 34 J=1.NKN
      INKN = NL + J
      PAK23(I.INKN)=RS24(I.J)
 34   CONTINUE
      DO 3   1.NL
      DO 3   1.NKN
      INKN = NL + J
      PAK33(I.INKN)=RS34(I.J)
   35 PAK33(INKN.I)=PAK33(I.INKN)
      DO 36 I=1.NKN
      INKN = NL + I
      DO 36 J=1.NKN
      JNKN = NL + J
   36 PAK33(INKN.JNKN)=RS44(I.J)
      IF(NSETS-4) 302.96.302
  302 CONTINUE
      INNKM= NL + NKN
      DO 37 I=1.NN
```

```
      DO 37 J=1,NKN
      INKM = INNKM + J
      PAK13(I,INKM)=RS15(I,J)
37    CONTINUE
      DO 38 I=1,NM
      DO 38 J=1,NKM
      INKM = INNKM + J
      PAK23(I,INKM)=RS25(I,J)
38    CONTINUE
      DO 39 I=1,NL
      DO 39 J=1,NKM
      INKM= INNKM + J
      PAK33(I,INKM)=RS35(I,J)
   39 PAK33(INKM,I)=PAK33(I,INKM)
      DO 40 I=1,NKN
      INKN = NL + I
      DO 40 J=1,NKM
      INKM = INNKM + J
      PAK33(INKN,INKM)=RS45(I,J)
   40 PAK33(INKM,INKN)=PAK33(INKN,INKM)
      DO 41 I=1,NKM
      INKM = INNKM + I
      DO 41 J=1,NKM
      JNKM = INNKM + J
   41 PAK33(INKM,JNKM)=RS55(I,J)
   96 CONTINUE
      N345 = NL + NKN + NKM
      CALL MATIV(PAK33,N345,0,DETRM,ID,IOUT)
      DO 42 I=1,NN
      DO 42 J=1,N345
      DO 42 K=1,N345
   42 WORK4(I,J)=WORK4(I,J) + PAK13(I,K)*PAK33(K,J)
      DO 43 I=1,NN
      DO 43 J=1,NN
      DO 43 K=1,N345
   43 WORK1(I,J) = WORK1(I,J) + WORK4(I,K)*PAK13(J,K)
      DO 44 I=1,NN
      DO 44 J=1,NN
   44 RST11(I,J) = RS11(I,J) - WORK1(I,J)
      DO 45 I=1,NM
      DO 45 J=1,N345
      DO 45 K=1,N345
   45 WORK5(I,J) = WORK5(I,J) + PAK23(I,K)*PAK33(K,J)
      DO 46 I=1,NM
      DO 46 J=1,NM
      DO 46 K=1,N345
   46 WORK2(I,J) = WORK2(I,J) + WORK5(I,K)*PAK23(J,K)
      DO 47 I=1,NM
      DO 47 J=1,NM
   47 RST22(I,J) = RS22(I,J) - WORK2(I,J)
      DO 48 I=1,NN
      DO 48 J=1,N345
      DO 48 K=1,N345
   48 WORK6(I,J) = WORK6(I,J) + PAK13(I,K)*PAK33(K,J)
```

```
      DO 49 I=1,NN
      DO 49 J=1,NM
      DO 49 K=1,N345
   49 WORK3(I,J) = WORK3(I,J) + WORK6(I,K)*PAK23(J,K)
      DO 50 I=1,NN
      DO 50 J=1,NM
      RST12(I,J) = RS12(I,J) - WORK3(I,J)
50    CONTINUE
      WRITE(IOUT,101)
101   FORMAT(1H0,T10,16HTHE R - STAR SET )
      WRITE(IOUT,102) ((RST11(I,J),J=1,NN),I=1,NN)
      WRITE(IOUT,102) ((RST12(I,J),J=1,NM),I=1,NN)
      WRITE(IOUT,102) ((RST22(I,J),J=1,NM),I=1,NM)
  102 FORMAT(1H0,T10,5E20.6)
      RETURN
      END
```

```
C
      SUBROUTINE MATIU(A.N1.M1.DETRM.IO.IOUT)
      DIMENSION A(5.5).B(5.1).INDEX(5.3)
      EQUIVALENCE (IROW.JROW).(ICOLM.JCOLM).(AMAX.T.SWAPP)
      IOUT = 3
      OE=1.E0
      OZ=0.E0
      M=M1
      N=N1
      DETRM=OE
      DO 20 J=1.N
   20 INDEX(J.3)=OZ
      DO 550 I=1.N
      AMAX=OZ
      DO 105 J=1.N
      IF(INDEX(J.3)-1) 6C.105.60
   60 DO 100 K=1.N
      IF(INDEX(K.3)-1) 80.100.715
   80 IF(AMAX-ABS(A(J.K))) 85.100.100
   85 IROW=J
      ICOLM_K
      AMAX=ABS(A(J.K))
  100 CONTINUE
  105 CONTINUE
      INDEX(ICOLM.3)=INDEX(ICOLM.3)+1
      INDEX(I.1)=IROW
      INDEX(I.2)=ICOLM
      IF(IROW-ICOLM) 140.310.140
  140 DETRM=-DETRM
      DO 200 L=1.N
      SWAPP=A(IROW.L)
      A(IROW.L)=A(ICOLM.L)
  200 A(ICOLM.L)=SWAPP
      IF(M) 310.310.210
  210 DO 250 L=1.M
      SWAPP=B(IROW.L)
      B(IROW.L)=B(ICOLM.L)
  250 B(ICOLM.L)=SWAPP
  310 PIVOT=A(ICOLM.ICOLM)
      DETRM=DETRM*PIVOT
      A(ICOLM.ICOLM)=OE
      DO 350 L=1.N
  350 A(ICOLM.L)=A(ICOLM.L)/PIVOT
      IF(M) 380.380.360
  360 DO 370 L=1.M
  370 B(ICOLM.L)=B(ICOLM.L)/PIVOT
  380 DO 550 L1=1.N
      IF(L1-ICOLM) 400.550.400
  400 T=A(L1.ICOLM)
      A(L1.ICOLM)=OZ
      CO 450 L=1.N
  450 A(L1.L)=A(L1.L)-A(ICOLM.L)*T
      IF(M) 550.550.460
  460 DO 500 L=1.M
```

```
500 B(L1,L)=B(L1,L)-B(ICOLM,L)*T
550 CONTINUE
    DO 710 I=1,N
    L=N+1-I
    IF(INDEX(L,1)-INDEX(L,2)) 630,710,630
630 JROW=INDEX(L,1)
    JCOLM=INDEX(L,2)
    DO 705 K=1,N
    SWAPP=A(K,JROW)
    A(K,JROW)=A(K,JCOLM)
    A(K,JCOLM)=SWAPP
705 CONTINUE
710 CONTINUE
    DO 730 K=1,N
    IF(INDEX(K,3)-1) 715,720,715
715 ID=2
    GO TO 740
720 CONTINUE
730 CONTINUE
    ID=1
740 CONTINUE
    WRITE(IOUT,1055)
1055 FORMAT(1H0///T10,14HTHE INVERSE IS  )
    DO 833 I=1,N
833  WRITE(IOUT,101) (A(I,J),J=1,N)
101  FORMAT(1H0,T10,5E20.6)
     WRITE(IOUT,111) DETRM
111  FORMAT(1H0,T10,17HTHE DETERMINANT = ,E20.6)
     RETURN
     END
```

```
C
      SUBROUTINE MATIV(A,N1,M1,DETRM,ID,IOUT)
      DIMENSION A(20,20),B(20,1),INDEX(20,3)
      EQUIVALENCE (IROW,JROW),(ICOLM,JCOLM),(AMAX,T,SWAPP)
      DE=1.E0
      DZ=0.E0
      M=M1
      N=N1
      DETRM=DE
      DO 20 J=1,N
   20 INDEX(J,3)=DZ
      DO 550 I=1,N
      AMAX=DZ
      DO 105 J=1,N
      IF(INDEX(J,3)-1) 60,105,60
   60 DO 100 K=1,N
      IF(INDEX(K,3)-1) 80,100,715
   80 IF(AMAX-ABS(A(J,K))) 85,100,100
   85 IROW=J
      ICOLM=K
      AMAX=ABS(A(J,K))
  100 CONTINUE
  105 CONTINUE
      INDEX(ICOLM,3)=INDEX(ICOLM,3)+1
      INDEX(I,1)=IROW
      INDEX(I,2)=ICOLM
      IF(IROW-ICOLM) 140,310,140
  140 DETRM=-DETRM
      DO 200 L=1,N
      SWAPP=A(IROW,L)
      A(IROW,L)=A(ICOLM,L)
  200 A(ICOLM,L)=SWAPP
      IF(M) 310,310,210
  210 DO 250 L=1,M
      SWAPP=B(IROW,L)
      B(IROW,L)=B(ICOLM,L)
  250 B(ICOLM,L)=SWAPP
  310 PIVOT=A(ICOLM,ICOLM)
      DETRM=DETRM*PIVOT
      A(ICOLM,ICOLM)=DE
      DO 350 L=1,N
  350 A(ICOLM,L)=A(ICOLM,L)/PIVOT
      IF(M) 380,380,360
  360 DO 370 L=1,M
  370 B(ICOLM,L)=B(ICOLM,L)/PIVOT
  380 DO 550 L1=1,N
      IF(L1-ICOLM) 400,550,400
  400 T=A(L1,ICOLM)
      A(L1,ICOLM)=DZ
      DO 450 L=1,N
  450 A(L1,L)=A(L1,L)-A(ICOLM,L)*T
      IF(M) 550,550,460
  460 DO 500 L=1,M
  500 B(L1,L)=B(L1,L)-B(ICOLM,L)*T
```

```
 550 CONTINUE
     DO 710 I=1.N
     L=N+1-I
     IF(INDEX(L.1)-INDEX(L.2)) 630.710.630
 630 JROW=INDEX(L.1)
     JCOLM=INDEX(L.2)
     DO 705 K=1.N
     SWAPP=A(K.JROW)
     A(K.JROW)=A(K.JCOLM)
     A(K.JCOLM)=SWAPP
 705 CONTINUE
 710 CONTINUE
     DO 730 K=1.N
     IF(INDEX(K.3)-1) 715.720.715
 715 ID=2
     GO TO 740
 720 CONTINUE
 730 CONTINUE
     ID=1
 740 CONTINUE
     WRITE(IOUT.1055)
1055 FORMAT(1H0///T10.14HTHE INVERSE IS   )
     DO 833 I=1.N
833  WRITE(IOUT.101) (A(I.J).J=1.N)
101  FORMAT(1H0.T10.5E20.6)
     WRITE(IOUT.111) DETRM
111  FORMAT(1H0.T10.17HTHE DETERMINANT = .E20.6)
     RETURN
     END
```

```
C
      SUBROUTINE TRILH(A,N,TX,TY,MK,NN,OZ,ATINV,IOUT)
      DIMENSION A(10,10),B(10,10),P(10,10),Q(10,10),U(10,10),V(10,10),TX
     1(10,10),TY(10,10),ATINV(10,10)
      DDN = 1.E-3
      DO 9 I=1,N
      DO 9 J=1,N
      ATINV(I,J)=OZ
    9 B(I,J)=OZ
      DO 965 I=1,N
  965 B(I,I)=1.E0
      DO 10 I=1,N
      U(1,I)=A(1,I)
      V(1,I)=B(1,I)
      P(1,I)=U(1,I)/U(1,1)
   10 Q(1,I)=V(1,I)/U(1,1)
      I=1
      DO 11 K=2,N
   13 CONTINUE
      IF(I) 15,16,15
   15 DO 12 J=1,N
      A(K,J)=A(K,J)-P(I,K)*U(I,J)
      B(K,J)=B(K,J)-P(I,K)*V(I,J)
      U(K,J)=A(K,J)
      V(K,J)=B(K,J)
   12 CONTINUE
      I=I-1
      GO TO 13
   16 I=I+K
      DO 19 LJ=1,N
      IF(ABS(U(K,K))-DDN) 26,26,25
   25 P(K,LJ)=U(K,LJ)/U(K,K)
      Q(K,LJ)=V(K,LJ)/U(K,K)
      GO TO 19
   26 U(K,LJ)=OZ
      V(K,LJ)=OZ
      P(K,LJ)=OZ
      Q(K,LJ)=OZ
   19 CONTINUE
   11 CONTINUE
      DO 18 I=1,N
      USQRT=SQRT(ABS(U(I,I)))
      IF(USQRT-DDN) 18,18,1
    1 DO 18 J=1,N
      U(I,J)=U(I,J)/USQRT
      V(I,J)=V(I,J)/USQRT
   18 CONTINUE
      GO TO (91,92,91),NN
   91 CONTINUE
      WRITE(IOUT,100)
  100 FORMAT(1H0,T20,30HTHE TRIANGULAR MATRIX IS AS FOLLOWING   )
      DO 96 I=1,N
      IF(ABS(U(I,I))-DDN) 96,96,95
   95 WRITE(IOUT,101) (V(I,J),J=1,N)
```

```
 96 CONTINUE
    GO TO 97
 92 CONTINUE
    DO 991 I=1.N
    DO 991 J=1.N
    IF(ABS(U(I.J))-DON) 992.992.991
992 U(I.J)=DZ
991 CONTINUE
    WRITE(IOUT.225)
225 FORMAT(1H0.T10.16HTHE T' MATRIX IS )
    DO 20 I=1.N
    IF(ABS(U(I.I))-DON) 20.20.21
21  WRITE(IOUT.101) (U(I.J).J=1.N)
101 FORMAT(1H0.T10.5E20.6)
 20 CONTINUE
 97 MK=N
    I=1
    IF(ABS(U(I.I))-DON) 32.32.31
 31 LK=1
 41 CONTINUE
    GO TO (222.221.222).NN
221 DO 33 J=1.N
 33 TX(LK.J)=U(I.J)
    GO TO 40
222 CONTINUE
    DO 223 J=1.N
223 TX(LK.J)=V(I.J)
    GO TO 40
 32 I=I+1
    MK=MK-1
    IF(I-N) 53.53.37
 53 CONTINUE
    IF(ABS(U(I.I))-DON) 32.32.31
 40 I=I+1
    LK=LK+1
    IF(I-N) 36.36.37
 36 IF(ABS(U(I.I))-DON) 42.42.41
 42 I=I+1
    MK=MK-1
    IF(I-N) 50.50.37
 50 CONTINUE
    IF(ABS(U(I.I))-DON) 42.42.41
 37 CONTINUE
    IF(MK) 60.61.60
 60 CONTINUE
    DO 55 I=1.MK
    DO 55 J=1.N
 55 TY(J.I)=TX(I.J)
    IF(NN-3) 99.98.99
 98 CONTINUE
    DO 234 I=1.N
    DO 234 J=1.N
234 ATINV(I.J)=DZ
    DO 235 I=1.MK
```

```
      DO 235 J=1,MK
      DO 235 K=1,N
  235 ATINV(I,J)=ATINV(I,J) +TX(I,K)*TY(K,J)
      GO TO 99
61    WRITE(IOUT,103)
103   FORMAT(1H0,T30,48HTHE MAIN DIAGONAL ELEMENTS OF HTE MATRIX ARE ALL
     1 //T31,47H ZEROS. TERMINATE THE EXECUTION OF THE PROGRAM   )
      PAUSE 1111
   99 RETURN
      END
```

```
C
      SUBPOUTINE DEPOV(A.N.DETRM)
      DIMENSION A(5.5)
      K=2
      L=1
    1 DO 10 I=K.N
      RATIO=A(I.L)/A(L.L)
      DO 10 J=K.N
      A(I.J)=A(I.J)-A(L.J)*RATIO
   10 CONTINUE
      IF(K-N) 15.20.20
   15 L=K
      K=K+1
      GO TO 1
   20 DETRM=1.
      DO 21 I=1.N
      DETRM=DETRM*A(I.I)
   21 CONTINUE
      RETURN
      END
```

```
C
C
C     SUBROUTINE MULCR
C
C*********************************************************************
C*
C*    THIS SUBROUTINE MULCR IS TO CALCULATE THE CANONICAL-MULTIPLE
C*    CORRELATION.
C*    SUBROUTINES  PACKA. MATIV. MULTIEIGEN. AND MANUP ARE CALLED TO THI
C*    PURPOSE..
C*
C*********************************************************************
C
      SUBROUTINE MULCR
      DIMENSION PRHOT(5,5).RST11(5.5).RST12(5.20).RST22(20.20).RESUT(20.
     120).AMURO(5).A(25).RSTY(25)
      COMMON N1.N2.N3.N4.N5.DZ.DE.IOUT.ML.NSETS
      COMMON X(25)
      COMMON R11(5.5).R12(5.5).R13(5.5).R14(5.5).R15(5.5).R21(5.5).R22(5
     1.5).R23(5.5).R24(5.5).R25(5.5).R31(5.5).R32(5.5).R33(5.5).R34(5.5)
     2.R35(5.5)
      COMMON R41(5.5).R42(5.5).R43(5.5).R44(5.5).R45(5.5).R51(5.5).R52(5
     1.5).R53(5.5).R54(5.5).R55(5.5).PRHO(5.5)
      KL=1
   10 CONTINUE
      GO TO (1.2.3.4.5).KL
    1 CONTINUE
      CALL PACKA(R11.R12.R13.R14.R15.R22.R23.R24.R25.R33.R34.R35.R44.R45
     1.R55.RST11.RST22.RST12.N1.N2.N3.N4.N5.NSETS.IOUT)
C
C*********************************************************************
C*
C*    THE SUBROUTINE PACKA IS TO PACK THE ORIGINAL R-MATIRX INTO A
C*    NEW MATRIX FOR CALCULATING THE CANONICAL-MULTIPLE CORRELATION
C*    WITH THE R11*.R12*.R21* R22* AS THE INPUT MATRICES.
C*
C*********************************************************************
C
      NM=N2+N3+N4+N5
      CALL MATIV(RST22.NM.O.DETRM.ID.IOUT)
C
C*********************************************************************
C*
C*    THE SUBROUTINE MATIV IS CALLED TO FIND THE INVERSES OF R11* AND R2
C*
C*********************************************************************
C
      N = N1
      CALL MULT1(RST11.RST22.RST12.N.NM.RESUT.DZ.IOUT.KL)
      GO TO 50
    2 CONTINUE
      CALL PACKA(R22.R21.R23.R24.R25.R11.R13.R14.R15.R33.R34.R35.R44.R45
     1.R55.RST11.RST22.RST12.N2.N1.N3.N4.N5.NSETS.IOUT)
      NM = N1 + N3 + N4 + N5
```

```
      CALL MATIV(RST22.NM.0.DETRM.ID.IOUT)
      N=N2
      CALL MULT1(RST11.RST22.RST12.N.NM.RESUT.DZ.IOUT.KL)
      GO TO 50
    3 CONTINUE
      CALL PACKA(R33.R31.R32.R34.R35.R11.R12.R14.R15.R22.R24.R25.R44.R45
     1.R55.RST11.RST22.RST12.N3.N1.N2.N4.N5.NSETS.IOUT)
      NM=N1+N2+N4+N5
      CALL MATIV(RST22.NM.0.DETRM.ID.IOUT)
      N=N3
      CALL MULT1(RST11.RST22.RST12.N.NM.RESUT.DZ.IOUT.KL)
      GO TO 50
    4 CONTINUE
      CALL PACKA(R44.R41.R42.R43.R45.R11.R12.R13.R15.R22.R23.R25.R33.R35
     1.R55.RST11.RST22.RST12.N4.N1.N2.N3.N5.NSETS.IOUT)
      NM=N1+N2+N3+N5
      CALL MATIV(RST22.NM.0.DETRM.ID.IOUT)
      N=N4
      CALL MULT1(RST11.RST22.RST12.N.NM.RESUT.DZ.IOUT.KL)
      GO TO 50
    5 CONTINUE
      CALL PACKA(R55.R51.R52.R53.R54.R11.R12.R13.R14.R22.R23.R24.R33.R34
     1.R44.RST11.RST22.RST12.N5.N1.N2.N3.N4.NSETS.IOUT)
      NM=N1+N2+N3+N4
      CALL MATIV(RST22.NM.0.DETRM.ID.IOUT)
      N=N5
      CALL MULT1(RST11.RST22.RST12.N.NM.RESUT.DZ.IOUT.KL)
   50 CONTINUE
      DO 80 I=1.N
      DO 80 J=I.N
      IJ=(J*(J-1))/2 + I
   80 A(IJ)=RESUT(I.J)
      CALL EIGEN(A.RSTY.N.0)
C
C*********************************************************************
C*
C*   THE SUBROUTINE EIGEN IS FROM SSP WHICH WILL BE USED TO FIND THE EIG
C*   VALUE AND THE EIGEN-VECTOR FOR THE SYMMETRIC MATRIX.
C*
C*********************************************************************
C*
      RHO= SQRT(ABS(A(1)))
      WRITE(IOUT.101) KL.A(1)
  101 FORMAT(1H0.T10.3HTHE.I3.12H EIGEN-VALUE //T10.E20.6)
      WRITE(IOUT.120) (RSTY(I).I=1.N)
  120 FORMAT(1H0.T10.16HTHE EIGEN-VECTOR //(T10.5E20.6))
      WRITE(IOUT.112)
  112 FORMAT(1H0.T10.34HTHE CANONICAL-MULTIPLE CORRELATION ///)
      WRITE(IOUT.102) RHO
  102 FORMAT(1H0.T10.5HRHO = .E20.6)
      AMURO(KL)=RHO
      KL=KL+1
      IF(KL-NSETS) 10.10.99
   99 CONTINUE
```

```
      KLL=KL-1
      WRITE(IOUT,222) (AMURO(I),I=1,KLL)
  222 FORMAT(1H0,T10,41HTHE CANONICAL-MULTIPLE CORRELATION MATRIX //(T10
     1,5E20.6))
C
C*****************************************************************************
C*
C*    THE SUBROUTINE MANUP IS TO FIND THE NORMALIZED CORRELATIONS BY
C*    MULTIPLY EACH OF THE CANONICAL-PARTIAL BY ITS APPROPRIATE CANONICAL
C*    MULTIPLE CORRELATIONS. I.E. -RHO(I,J)/SQRT((1-AMURO(I)**2)*(1-AMURO
C*
C*****************************************************************************
C
      CALL MANUP(PRHO,AMURO,KLL,IOUT)
      DO 1119 I=1,KLL
      DO 1119 J=1,KLL
      PRHOT(I,J)=PRHO(I,J)
 1119 CONTINUE
      CALL MATIU(PRHOT,KLL,0,DETRM,ID,IOUT)
      CALL PADJS(PRHO,KLL,IOUT)
 1132 CALL RHOCT
 1133 RETURN
      END
```

```
C
      SUBROUTINE PACKA(RT11,RT12,RT13,RT14,RT15,RT22,RT23,RT24,RT25,RT33
     1,RT34,RT35,RT44,RT45,RT55,RST11,RST22,RST12,N,M,L,KN,KM,NSETS,IOUT
     2)
      DIMENSION RT11(5,5),RT12(5,5),RT13(5,5),RT14(5,5),RT15(5,5),RT22(5
     1,5),RT23(5,5),RT24(5,5),RT25(5,5),RT33(5,5),RT34(5,5),RT35(5,5),RT
     144(5,5),RT45(5,5),RT55(5,5)
      DIMENSION RST11(5,5),RST12(5,20),RST22(20,20)
      DO 10 I=1,N
      DO 10 J=1,N
   10 RST11(I,J)=RT11(I,J)
      DO 11 I=1,N
      DO 11 J=1,M
      RST12(I,J)=RT12(I,J)
 11   CONTINUE
      DO 13 I=1,M
      DO 13 J=1,M
   13 RST22(I,J)=RT22(I,J)
      IF(NSETS - 2) 300,96,300
 300  CONTINUE
      DO 12 I=1,N
      DO 12 J=1,N
      JJ = M + J
      RST12(I,JJ) = RT13(I,J)
 12   CONTINUE
      DO 14 I=1,M
      DO 14 J=1,L
      JJ = M + J
      RST22(I,JJ)=RT23(I,J)
 14   RST22(JJ,I) = RST22(I,JJ)
      DO 15 I=1,L
      II = M + I
      DO 15 J=1,L
      JJ = M + J
   15 RST22(II,JJ)=RT33(I,J)
      MM = M + L
      IF(NSETS-3) 301,96,301
  301 CONTINUE
      DO 16 I=1,N
      DO 16 J=1,KN
      JKN = MM + J
      RST12(I,JKN)=RT14(I,J)
   16 RST12(JKN,I)=RST12(I,JKN)
      DO 17 I=1,M
      DO 17 J=1,KN
      JKN = MM + J
      RST22(I,JKN)=RT24(I,J)
   17 RST22(JKN,I)=RST22(I,JKN)
      DO 18 I=1,L
      II = M + I
      DO 18 J=1,KN
      JJ = MM + J
      RST22(II,JJ)=RT34(I,J)
   18 RST22(JJ,II)=RST22(II,JJ)
```

```
      MM = M + L + KN
      IF(NSETS-4) 302,96,302
  302 CONTINUE
      DO 118 I=1,N
      DO 118 J=1,KM
      JKM = MM + J
      RST12(I,JKM)=RT15(I,J)
 118  CONTINUE
      DO 19 I=1,M
      DO 19 J=1,KM
      JKM = MM + J
      RST22(I,JKM)=RT25(I,J)
   19 RST22(JKM,I)=RST22(I,JKN)
      DO 20 I=1,L
      II = M + I
      DO 20 J=1,KM
      JJKM = MM + J
      RST22(II,JJKM)=RT35(I,J)
   20 RST22(JJKM,II)=RST22(JJKM,II)
      DO 21 I=1,KN
      II = M + L + I
      DO 21 J=1,KM
      JJKM = MM + J
      RST22(II,JJKM)=RT45(I,J)
   21 RST22(JJKM,II)=RST22(II,JJKM)
      DO 22 I 1,KM
      II = MM + I
      DO 22 J=1,KM
      JJ = MM + J
   22 RST22(II,JJ)=RT55(I,J)
   96 CONTINUE
      WRITE(IOUT,102)
 102  FORMAT(1H0,T10,41HTHE CANONICAL-MULTIPLE CORRELATION MATRIX )
      MM = M + L
      IF(NSETS-3) 303,97,303
  303 MM = M + L + KN
      IF(NSETS-4) 304,97,304
  304 MM = M + L + KN + KM
   97 CONTINUE
      WRITE(IOUT,101) ((RST11(I,J),J=1,N),I=1,N)
      WRITE(IOUT,101) ((RST12(I,J),J=1,MM),I=1,N)
      WRITE(IOUT,101) ((RST22(I,J),J=1,MM),I=1,MM)
  101 FORMAT(1H0,T10,5E20.6)
      RETURN
      END
```

```
C
      SUBROUTINE MANUP(PRHO,AMURO,KLL,IOUT)
      DIMENSION PRHO(5,5),AMURO(5)
      DE=1.E0
      DO 10 I=1,KLL
      PRHO(I,I)=PRHO(I,I)/(DE-AMURO(I)**2)
      DO 10 J=1,KLL
      IF(I-J) 11,10,11
   11 PRHO(I,J)=-PRHO(I,J)/SQRT((DE-AMURO(I)**2)*(DE-AMURO(J)**2))
   10 CONTINUE
      WRITE(IOUT,100)
  100 FORMAT(1H0,T10,26HTHE NORMALIZED CORRELATION   )
      DO 12 I=1,KLL
      WRITE(IOUT,101) (PRHO(I,J),J=1,KLL)
  101 FORMAT(1H0,T10,5E20.6)
   12 CONTINUE
      RETURN
      END

C
      SUBROUTINE PARHO(PRHO,IN,IOUT)
      DIMENSION PRHO(5,5)
      DO 10 I=1,IN
   10 PRHO(I,I)=1.E0
      IIN=IN-1
      DO 61 I=1,IIN
      II=I+1
      DO 61 J=II,IN
   61 PRHO(J,I)=PRHO(I,J)
      WRITE(IOUT,108)
  108 FORMAT(1H0,T10,28HTHE CANONICAL-PARTIAL MATRIX )
      DO 62 I=1,IN
      WRITE(IOUT,109) (PRHO(I,J),J=1,IN)
  109 FORMAT(1H0,T10,5E20.6)
   62 CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE MULT2(TX.RST22.RST12.N.MK.M.RESUT.DZ.IOUT.KL)
      DIMENSION TX(10.10).RST22(20.20).RST12(10.20).RESUT(20.20)
      DIMENSION WORK1(10.20).WORK2(20.20)
      DATA WORK1.WORK2/500*0.E0/
      DO 1111 I=1.10
      DO 1111 J=1.10
 1111 RESUT(I.J)=0.E0
      DO 11 I=1.MK
      DO 11 J=1.M
      DO 11 K=1.N
   11 WORK1(I.J)=WORK1(I.J)+TX(I.K)*RST12(K.J)
      DO 12 I=1.MK
      DO 12 J=1.M
      DO 12 K=1.M
   12 WORK2(I.J)=WORK2(I.J)+WORK1(I.K)*RST22(K.J)
      DO 14 I=1.MK
      DO 14 J=1.MK
      DO 14 K=1.M
   14 RESUT(I.J)=RESUT(I.J)+WORK2(I.K)*WORK1(J.K)
      WRITE(IOUT.101) KL
 101  FORMAT(1H0.T10.3HTHE.I3.42H MATRIX FOR CANONICAL-PARTIAL CORRELATI
     1ON )
      DO 15 I=1.MK
      WRITE(IOUT.102) (RESUT(I.J).J=1.MK)
  102 FORMAT(1H0.T10.5E20.6)
   15 CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE MULT1(RST11.RST22.RST12.N.M.RESUT.DZ.IOUT.KL)
      DIMENSION RST11(5.5).RST22(20.20).RST12(5.20).RESUT(20.20)
      DIMENSION WORK1(5.20).WORK2(20.20)
      DATA WORK1.WORK2/500*0.E0/
      DO 10 I=1,10
      DO 10 J=1,10
   10 RESUT(I.J)=DZ
      DO 11 I=1.N
      DO 11 J=1.M
      DO 11 K=1.N
   11 WORK1(I.J)=WORK1(I.J)+RST11(I.K)*RST12(K.J)
      DO 12 I=1.N
      DO 12 J=1.M
      DO 12 K=1.M
   12 WORK2(I.J)=WORK2(I.J)+WORK1(I.K)*RST22(K.J)
      DO 14 I=1.N
      DO 14 J=1.N
      DO 14 K=1.M
14    RESUT(I.J) = RESUT(I.J) + WORK2(I.K)*RST12(J.K)
      WRITE(IOUT.101) KL
101   FORMAT(1H0.T10.3HTHE.I3.42H MATRIX FOR CANONICAL-MULTIPLE CORRELAT
     1ION  )
      DO 30 I=1.N
30    WRITE(IOUT.102) (RESUT(I.J).J=1.N)
102   FORMAT(1H0.T10.5E20.6)
      RETURN
      END
```

```
C
      SUBROUTINE RHOCL(RHO.X.XX.R12.R13.R14.R15.R23.R24.R25.R34.R35.R45.
     1MK1.MK2.MK3.MK4.MK5.IEND)
      DIMENSION X(25).XX(25).R12(5.5).R13(5.5).R14(5.5).R15(5.5).R23(5.5
     1).R24(5.5).R25(5.5).R34(5.5).R35(5.5).R45(5.5).RHO(5.5)
      M2=MK1+MK2
      M3 = M2 + MK3
      M4 = M3 + MK4
      DO 30 K=1.MK2
      MI1=MK1+K
      DO 30 KK=1.MK1
      RHO(1.2)=RHO(1.2)+X(KK)*R12(KK.K)*X(MI1)/(XX(1)*XX(2))
   30 CONTINUE
      IF(IEND - 2) 89.91.89
 89   CONTINUE
      DO 31 KI=1.MK3
      MI2=M2+KI
      DO 29 KK=1.MK1
      RHO(1.3)=RHO(1.3)+X(KK)*R13(KK.KI)*X(MI2)/(XX(1)*XX(3))
   29 CONTINUE
      DO 28 K=1.MK2
      MI1=MK1+K
      RHO(2.3)=RHO(2.3)+X(MI1)*R23(K.KI)*X(MI2)/(XX(2)*XX(3))
   28 CONTINUE
   31 CONTINUE
      IF(IEND - 3) 90.91.90
 90   DO 41 K=1.MK4
      MI3 = M3 + K
      DO 42 KK=1.MK1
 42   RHO(1.4) = RHO(1.4) + X(KK)*R14(KK.K)*XX(MI3)/(XX(1)*XX(4))
      DO 43 KK=1.MK2
      MI1 = MK1 + KK
 43   RHO(2.4) = RHO(2.4) + X(MI1)*R24(KK.K)*X(MI3)/(XX(2)*XX(4))
      DO 44 KK=1.MK3
      MI2 = M2 + KK
 44   RHO(3.4) = RHO(3.4) + X(MI2)*R34(KK.K)*X(MI3)/(XX(3)*XX(4))
 41   CONTINUE
      IF(IEND - 4) 92.91.92
 92   DO 51 K=1.MK5
      MI4 = M4 + K
      DO 52 KK=1.MK1
 52   RHO(1.5) = RHO(1.5) + X(KK)*R15(KK.K)*X(MI4)/(XX(1)*XX(5))
      DO 53 K=1.MK2
      MI1 = MK1 + KK
 53   RHO(2.5) = RHO(2.5) + X(MI1)*R25(KK.K)*X(MI4)/(XX(2)*XX(5))
      DO 54 KK=1.MK3
      MI2 = M2 + KK
 54   RHO(3.5) = RHO(3.5) + X(MI2)*R35(KK.K)*X(MI4)/(XX(3)*XX(5))
      DO 55 KK=1.MK4
      MI3 = M3 + KK
 55   RHO(4.5) = RHO(4.5) + X(MI3)*R45(KK.K)*X(MI4)/(XX(4)*XX(5))
 51   CONTINUE
 91   CONTINUE
      DO 32 I=1.IEND
```

```
      DO 32 J=1.IEND
      IF(I-J) 34.32,34
   34 RHO(J.I)=RHO(I.J)
   32 CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE RHOCT
      DIMENSION XX(25).RHO(5.5)
      COMMON N1.N2.N3.N4.N5.DZ.DE.IOUT.ML.IEND
      COMMON X(25)
      COMMON R11(5.5).R12(5.5).R13(5.5).R14(5.5).R15(5.5).R21(5.5).R22(5
     1.5).R23(5.5).R24(5.5).R25(5.5).R31(5.5).R32(5.5).R33(5.5).R34(5.5)
     1.R35(5.5).R41(5.5).R42(5.5).R43(5.5).R44(5.5).R45(5.5).R51(5.5).R5
     22(5.5).R53(5.5).R54(5.5).R55(5.5).PRHO(5.5)
      WRITE(IOUT.1113)
 1113 FORMAT(1H1)
      WRITE(IOUT.101)
 101  FORMAT(1H0.T10.42HTHE INPUT WEIGHTS FROM FLETCHER AND POWELL )
      WRITE(IOUT.100) (X(I).I=1.ML)
  100 FORMAT(1H0.T11.5E20.6)
      DO 920 I=1.IEND
      DO 920 J=1.IEND
 920  RHO(I.J)=DZ
      DO 9 I=1.IEND
    9 RHO(I.I)=DE
      DO 10 I=1.ML
   10 XX(I)=DZ
      DO 11 I=1.N1
   11 XX(1)=XX(1)+X(I)**2
      DO 12 I=1.N2
      II=N1+I
   12 XX(2)=XX(2)+X(II)**2
      DO 13 I=1.N3
      II=N1+N2+I
   13 XX(3)=XX(3)+X(II)**2
      DO 15 I=1.N4
      II = N1 + N2 + N3 + I
 15   XX(4) = XX(4) + X(II)**2
      DO 16 I=1.N5
      II = N1 + N2 + N3 + N4 + I
 16   XX(5) = XX(5) + X(II)**2
      DO 14 I=1.IEND
   14 XX(I)=SQRT(XX(I))
      CALL RHOCL(RHO.X.XX.R12.R13.R14.R15.R23.R24.R25.R34.R35.R45.RN1.N2
     1.N3.N4.N5.IEND)
      WRITE(IOUT.112)
 112  FORMAT(1H0.T10.14HTHE INPUT DATA )
      WRITE(IOUT.113)((R11(I.J).J=1.N1).I=1.N1)
      WRITE(IOUT.113)((R12(I.J).J=1.N2).I=1.N1)
      WRITE(IOUT.113)((R22(I.J).J=1.N2).I=1.N2)
      IF(IEND - 2) 888.90.888
 888  CONTINUE
      WRITE(IOUT.113)((R13(I.J).J=1.N3).I=1.N1)
      WRITE(IOUT.113)((R23(I.J).J=1.N3).I=1.N2)
      WRITE(IOUT.113)((R33(I.J).J=1.N3).I=1.N3)
  113 FORMAT(1H0.T10.5E20.6)
      IF(IEND - 3) 91.90.91
 91   WRITE(IOUT.113) ((R14(I.J).J=1.N4).I=1.N1)
      WRITE(IOUT.113) ((R24(I.J).J=1.N4).I=1.N2)
```

```
      WRITE(IOUT,113) ((R34(I,J),J=1,N4),I=1,N3)
      WRITE(IOUT,113) ((R44(I,J),J=1,N4),I=1,N4)
      IF(IEND - 4) 92,90,92
92    WRITE(IOUT,113) ((R15(I,J),J=1,N5),I=1,N1)
      WRITE(IOUT,113) ((R25(I,J),J=1,N5),I=1,N2)
      WRITE(IOUT,113) ((R35(I,J),J=1,N5),I=1,N3)
      WRITE(IOUT,113) ((R45(I,J),J=1,N5),I=1,N4)
      WRITE(IOUT,113) ((R55(I,J),J=1,N5),I=1,N5)
90    CONTINUE
      WRITE(IOUT,110)
110   FORMAT(1H0,T10,35HTHE CANONICAL-WEIGHTED CORRELATION )
      DO 3030 I=1,IEND
3030  WRITE(IOUT,111) (RHO(I,J),J=1,IEND)
 111  FORMAT(1H0,//(T10,5E20.6))
      CALL PADJ6(RHO,IEND,IOUT)
      CALL MATIU(RHO,IEND,0,DETRM,ID,IOUT)
      WRITE(IOUT,115) DETRM
115   FORMAT(1H0,T10,25HTHE DETERMINANT OF RHO IS ,E20.6)
      RETURN
      END
```

```
      SUBROUTINE PAOJ6(PRHO,KLL,IOUT)
      DIMENSION PRHO(5,5),PRHOT(5,5),PRHOW(5,5)
      K = 1
      DO 40 I=1,KLL
      DO 40 J=1,KLL
      PRHOW(I,J) = PRHO(I,J)
40    PRHOT(I,J) = PRHO(I,J)
39    CONTINUE
      WRITE(IOUT,101)
101   FORMAT(1H0,T10,28HTHE INPUT CORRELATION MATRIX  )
      DO 444 I=1,KLL
      WRITE(IOUT,108) (PRHOT(I,J),J=1,KLL)
108   FORMAT(1H0,T10,5E20.6)
444   CONTINUE
      CALL DEPOV(PRHOW,KLL,DETM)
      WRITE(IOUT,102) DETM
102   FORMAT(1H0,T10,27HTHE DETERMINANT OF MATRIX =    ,E20.6)
      CALL MATIS(PRHOT,KLL,0,DETRM,IO,IOUT)
      DETEM = 1./DETRM
      WRITE(IOUT,555) DETEM
555   FORMAT(1H0,T10,33HTHE INVERSE OF THE DETERMINANT IS    ,E20.6)
      GO TO (77,1,2,3,4,5,6,7,8,9,99),K
77    CONTINUE
      DO 55 I=1,KLL
      DO 55 J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
      IF(I - J) 56,55,56
56    PRHOT(I,J) = -PRHOT(I,J)
55    CONTINUE
      DO 57 I=1,KLL
      DO 57 J=1,KLL
57    PRHOW(I,J) = PRHOT(I,J)
      K = K + 1
      GO TO 39
1     DO 41 I=1,KLL
      DO 41 J=1,KLL
      PRHOW(I,J) =  ABS(PRHO(I,J))
41    PRHOT(I,J) =  ABS(PRHO(I,J))
      K = K + 1
      GO TO 39
2     DO 42 I=1,KLL
      DO 42 J=1,KLL
      PRHOW(I,J) =   ABS(PRHO(I,J))
42    PRHOT(I,J) =  ABS(PRHO(I,J))
      PRHOT(1,2) = -PRHOT(1,2)
      PRHOT(2,1) = -PRHOT(2,1)
      PRHOW(1,2) = PRHOT(1,2)
      PRHOW(2,1) = PRHOT(2,1)
      K = K + 1
      GO TO 39
3     DO 43 I=1,KLL
      DO 43 J=1,KLL
      PRHOT(I,J) =  ABS(PRHO(I,J))
43    PRHOW(I,J) = PRHOT(I,J)
```

```
      PRHOT(1,3) =  -PRHOT(1,3)
      PRHOT(3,1) = -PRHOT(3,1)
      PRHOW(1,3) = PRHOT(1,3)
      PRHOW(3,1) = PRHOT(3,1)
      K = K + 1
      GO TO 39
4     DO 44 I=1,KLL
      DO 44 J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
44    PRHOW(I,J) = ABS(PRHO(I,J))
      PRHOT(2,3) = -PRHOT(2,3)
      PRHOT(3,2) = -PRHOT(3,2)
      PRHOW(2,3) = PRHOT(2,3)
      PRHOW(3,2) = PRHOT(3,2)
      K = K + 1
      GO TO 39
5     DO 45 I=1,KLL
      DO 45 J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
45    PRHOW(I,J) = ABS(PRHO(I,J))
      PRHOT(1,4) = -PRHOT(1,4)
      PRHOT(4,1) = -PRHOT(4,1)
      PRHOW(1,4) = PRHOT(1,4)
      PRHOW(4,1) = PRHOT(4,1)
      K = K + 1
      GO TO 39
6     DO 46 I=1,KLL
      DO 46 J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
46    PRHOW(I,J) = ABS(PRHO(I,J))
      PRHOT(2,4) = -PRHOT(2,4)
      PRHOT(4,2) = -PRHOT(4,2)
      PRHOW(2,4) = PRHOT(2,4)
      PRHOW(4,2) = PRHOT(4,2)
      K = K + 1
      GO TO 39
7     DO 47 I=1,KLL
      DO 47 J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
47    PRHOW(I,J) = ABS(PRHO(I,J))
      PRHOT(3,4) = -PRHOT(3,4)
      PRHOT(4,3) = -PRHOT(4,3)
      PRHOW(3,4) = PRHOT(3,4)
      PRHOW(4,3) = PRHOT(4,3)
      K = K + 1
      GO TO 39
8     DO 48 I=1,KLL
      DO 48  J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
48    PRHOW(I,J) = ABS(PRHO(I,J))
      PRHOT(1,2) = -PRHOT(1,2)
      PRHOT(2,1) = -PRHOT(2,1)
      PRHOT(2,4) = -PRHOT(2,4)
      PRHOT(4,2) = -PRHOT(4,2)
```

```
      PRHOW(1.2) = PRHOT(1.2)
      PRHOW(2.1) = PRHOT(2.1)
      PRHOW(2.4) = PRHOT(2.4)
      PRHOW(4.2) = PRHOT(4.2)
      K = K + 1
      GO TO 39
9     DO 49 I=1,KLL
      DO 49 J=1,KLL
      PRHOT(I,J) = ABS(PRHO(I,J))
49    PRHOW(I,J) = ABS(PRHO(I,J))
      PRHOT(1.2) = -PRHOT(1.2)
      PRHOT(2.1) = -PRHOT(2.1)
      PRHOT(3.4) = -PRHOT(3.4)
      PRHOT(4.3) = -PRHOT(4.3)
      PRHOW(1.2) = PRHOT(1.2)
      PRHOW(2.1) = PRHOT(2.1)
      PRHOW(3.4) = PRHOT(3.4)
      PRHOW(4.3) = PRHOT(4.3)
      K = K + 1
      GO TO 39
99    RETURN
      END
```

APPENDIX E

COMPUTER PROGRAM FOR THE REDUCTION SCALES

FOR E MATRIX

"R T E"

```
C
C  THE PROGRAM FOR FINDING NEW WEIGHTS FOR E
C
      DIMENSION NRST(5).X(20).NT(5.5).T(5.5.5).XDEX(5.5).IROW(5)
     1 . PK(5).PL(5).W(5.5).WSTR1(5.5).WSTR2(5.5)
      DATA W.WSTR1.WSTR2.PL.PK/85*0.E0/
C
      REWIND 10
      IOUT = 6
      INPUT = 10
      READ(INPUT.101) NWTS.NWT1.WT2.WT3
 101  FORMAT(2I5.2E20.6)
      READ(INPUT.102) NSETS.(NRST(I).I=1.5)
 102  FORMAT(6I2)
      READ(INPUT.103) (X(I).I=1.NWTS)
 103  FORMAT(4E20.6)
      DO 10 JA=1.NSETS
      IA = NRST(JA) + 1
      READ(INPUT.104) (NT(JA.I).I=1.IA)
 104  FORMAT(5I8)
 10   CONTINUE
      READ(INPUT.104) NTAL
      DO 11 J=1.NSETS
      READ(INPUT.105) NO.MK.((T(J.IA.JA).JA=1.NO).IA=1.MK)
 105  FORMAT(2I2.4E19.6)
C105  FORMAT(2I2.4E20.6)
      IROW(J) = MK
 11   CONTINUE
      IA = 0
      DO 12 I=1.NSETS
      JA = NRST(I)
      DO 12 JB=1.JA
      IA = IA + 1
      XDEX(I.JB) = X(IA)
 12   CONTINUE
      DO 20 ISET=1.NSETS
      MK = IROW(ISET)
      NO = MK + 1
      DO 21 IX=1.MK
      DO 21 JX=1.NO
      W(ISET.JX) = W(ISET.JX) + XDEX(ISET.IX)*T(ISET.IX.JX)
 21   CONTINUE
 20   CONTINUE
      DO 23 ISET=1.NSETS
      MENT = NRST(ISET) + 1
      DO 24 J=1.MENT
 24   PK(ISET) =PK(ISET) + FLOAT(NT(ISET.J))*W(ISET.J)/FLOAT(NTAL)
      DO 28 J=1.MENT
 28   WSTR1(ISET.J) = W(ISET.J) - PK(ISET)
 23   CONTINUE
      DO 25 ISET=1.NSETS
      MENT = NRST(ISET) + 1
      DO 26 J=1.MENT
 26   PL(ISET) = PL(ISET) + FLOAT(NT(ISET.J))*WSTR1(ISET.J)*WSTR1(ISET.J
```

```
     1)/FLOAT(NTAL)
25    PL(ISET) =  SQRT(PL(ISET))
      DO 29 ISET=1,NSETS
      MENT = NRST(ISET) + 1
      DO 30 J=1,MENT
30    WSTR2(ISET,J) = WSTR1(ISET,J)/PL(ISET)
29    CONTINUE
      WRITE(IOUT,151)
151   FORMAT(1H1/////T35,59HTHE PROGRAM FOR FINDING THE CATEGORICAL SCAL
     1ES FOR E_MATRIX    )
      WRITE(IOUT,152) NSETS,(NRST(I),I=1,5)
152   FORMAT(1H0,T10,34HNUMBER OF SETS (NSETS)          =,I5//
     r           T10,34HNUMBER OF ROWS IN (1,1) SET (N1) =,I5//
     r           T10,34HNUMBER OF ROWS IN (2,2) SET (N2) =,I5//
     r           T10,34HNUMBER OF ROWS IN (3,3) SET (N3) =,I5//
     r           T10,34HNUMBER OF ROWS IN (4,4) SET (N4) =,I5//
     r           T10,34HNUMBER OF ROWS IN (5,5) SET (N5) =,I5//       )
      WRITE(IOUT,153)
153   FORMAT(1H0//T10,44HTHE WEIGHTS FROM FLETCHER_AND POWELL PROGRAM__ )
      DO 171 I=1,NWTS
      WRITE(IOUT,154) I,X(I)
154   FORMAT(1H0,T20,2HX(,I3,4H_) =,E20.6)
171   CONTINUE
      DO 173 JA=1,NSETS
      IA = NRST(JA) +1
      WRITE(IOUT,172) JA,(NT(JA,I),I=1,IA)
172   FORMAT(1H0,T10,26HTHE_MARGINAL TOTAL FOR SET ,I3//(T10,5I8))
173   CONTINUE
      WRITE(IOUT,158) NTAL
159   FORMAT(1H0,T10,17HTHE GRAND TOTAL =,I8)
      WRITE(IOUT,160)
160   FORMAT(1H0,T10,26HTHE T CONDITIONAL INVERSES  )
      DO 175 J=1,NSETS
      MK = IROW(J)
      NO = MK + 1
      WRITE(IOUT,180) J
180   FORMAT(1H0,T10, 8HFOR  SET ,I3)
      DO 181 I=1,MK
181   WRITE(IOUT,182) (T(J,I,JA),JA=1,NO)
182   FORMAT(1H0,T10,5E20.6)
175   CONTINUE
      WRITE(IOUT,185)
185   FORMAT(1H0/////T10,40HTHE CATEGORICAL WEIGHTS FOR THE E MATRIX  )
      DO 186 ISET=1,NSETS
      WRITE(IOUT,161) ISET
161   FORMAT(1H0//T10,31HTHE CATEGORICAL WEIGHTS FOR SET  ,I3)
      MENT = NRST(ISET) + 1
      WRITE(IOUT,189) (WSTR2(ISET,J),J=1,MENT)
189   FORMAT(1H0,T10,5E20.6)
186   CONTINUE
      CALL EXIT
      END
```